Bayesian model-based clustering

Fuentes Garcia, Ruth S.

Award date:
2004

Awarding institution:
University of Bath

Bayesian model-based clustering

submitted by

Ruth S. Fuentes García

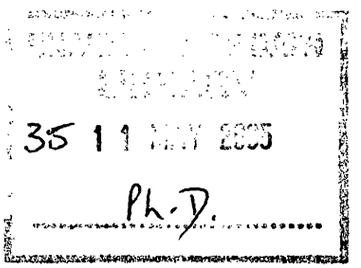
for the degree of Ph.D.

of the

University of Bath

2004

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346


35 11 MAY 2005
Ph.D.

To UNAM – its hard work keeps good education accessible for many young Mexicans.

Abstract

Clustering of multivariate data is an established problem of wide application and has been of increasing importance as large and complex data sets have become more commonplace. Model-based clustering, usually employing mixtures of multivariate Gaussian distributions, has recently received a great deal of attention. Particularly attractive is the probabilistic interpretation of the model and the potential for classification of future observations.

Recent advances in Markov chain Monte Carlo (MCMC) have made the Bayesian approach to mixture modelling rather attractive. In this thesis, we concentrate on the use of Gaussian finite mixture models to identify clusters in multivariate data sets. The possibility of having Markov transitions between states of different dimension in *trans-dimensional* samplers allows us to consider the number of components as an unknown parameter. Hence, no definite information on the number of groups is required. We present the results obtained for several multivariate data sets when clusters are found through the use of the reversible jump MCMC samplers, Green [32] and Richardson and Green [51]), and the birth-and-death MCMC sampler Stephens [59].

We emphasize the fact that, in practical clustering, the description of one group by only one component of the mixture model has proved to be ambitious in many cases. It is often difficult to describe the behavior of data that belong to the same group with a single multivariate normal distribution, or in fact any multivariate distribution. One suggestion is to attempt to represent a complex cluster structure as itself a mixture of standard normal components. To follow this line we restrict the shapes of the normal distributions, allowing simpler parametric forms for each component. A cluster would be formed by a submixture of components of the resulting fitted mixture. To define the submixtures that comprise a single cluster the sampled values for the parameters will be post-processed and criteria for submixture membership applied.

The groups in the data could then be defined combining the information obtained from the general mixtures of multivariate normal distributions and the submixture model.

Acknowledgements

I would like to thank my supervisor Dr. Tony Robinson for his friendship, help and advice. His careful revision of my work always encouraged the development of this project.

I wish to thank the *National Council for Science and Technology* (CONACYT), México, for its financial support: scholarship 149104/154942.

I would like to thank all the people from the Department of Mathematical Sciences at the University of Bath, for their everyday help.

Last but not least, all my gratitude to Ramsés, my family, and all my friends for their love and support.

CONTENTS

CHAPTER 1

Introduction

1.1 Cluster analysis

Large multivariate data sets arise in many areas of research. Part of the process to obtain useful information from the collected data is to identify the existence of natural groups. The latter is the main objective of cluster analysis, a widely used statistical tool. There is a large amount of literature on cluster analysis and classification methods. Clustering and classification are not described as a well-integrated subject and the concept of cluster itself is not very easy to state. Broadly speaking, we are interested in the allocation of n observations into k groups or clusters, where each observation belongs to one and only one group. According to some criterion, the elements that belong to the same group are similar and they are dissimilar to elements from other groups. In other words, we are looking for compact groups which at the same time are isolated from other groups.

Cluster analysis has developed through a collection of methods and there are several issues that make it a difficult problem. Many clustering methods impose some structure to the data and search through the sets of all possible clustering configurations to find the one that is optimum in terms of some criterion. Therefore, it is possible to find that different methodologies lead to different conclusions. When a clustering algorithm is applied to a data set, a classification is obtained, even when the data do not exhibit a natural grouping structure. Spurious groups could be found even in random data sets, in this context, finding real groups is an important problem. Measures of similarity and dissimilarity are required by some methodologies, other methods require the number of groups to be defined before partitioning the data, which is a major problem in cluster

analysis. The researcher often ignores the number of groups present in the population. All these are aspects that often result in techniques that are not based in probability models and are difficult to evaluate.

Few of the techniques are based on the use of a probability model in a cluster analysis context, which was found useful as both a new method and a way to understand when the existing methodologies were likely to be successful. Currently there is a lot of interest in model-based clustering and the literature available is vast over this and related topics. We present a review of some of these papers relevant to our work in chapter 2.

In this thesis, we will consider a Bayesian approach to model-based clustering using a finite mixture of Gaussian distributions via Markov chain Monte Carlo (MCMC). In a model-based clustering approach, the data itself will be allowed to determine the characteristics and the number of the clusters. Therefore the number of components of the mixture model will be considered as an unknown parameter subject to estimation. As a result, the use of MCMC sampling methods which allow a change in dimension will be essential. This approach will give us some assessment of the uncertainty of the clustering results, particularly for the number of clusters. We are interested in learning about the performance of different sampling methods and the adequacy of the model for a multivariate clustering problem.

In this chapter we introduce some general concepts required to follow the development of this work. In the following section, we present a general overview of the clustering methods which are widely used. For a thorough presentation of these methodologies see for example Hartigan [37], Gordon [31], Seber [58].

1.1.1 Some clustering methodologies

The input data given to clustering methods could be of different classes such as p-dimensional vectors of data, proximity matrices and/or similarity (dissimilarity) matrices. The algorithms that search through all clustering configurations are divided in Hartigan [37] as:

- *sorting* type, where the data are sorted and partitioned by means of only one variable;
- *switching* type, where after a partition, observations could be reallocated into a

different cluster;

- *joining* type, where near clusters are fused to form a new one;
- *splitting* type, where further clusters are divided;
- *addition* type, where a set of clusters is already available and each object is added in turn;
- *searching* type, where it is feasible to search through all eligible cluster structures to select the one that satisfies an optimality condition.

Many similarity measures have been used in the literature. If we have a population of objects $\mathcal{P}$, a similarity can be defined as a function that maps $\mathcal{P} \times \mathcal{P}$ into $\mathbb{R}$ satisfying:

$$\begin{aligned} 0 \leq \ & c_{rs} \ \leq 1 \quad \forall \quad r, s \in \mathcal{P} \\ & c_{rr} \ = \ 1 \\ & c_{rs} \ = \ 1 \quad \text{if and only if} \quad r = s \\ & c_{rs} \ = \ c_{sr}. \end{aligned}$$

It has been argued that order relationships are more important than numerical values, so there is no need for the similarity measure to be a metric. A dissimilarity measure can be defined from a similarity measure by considering $d_{rs} = 1 - c_{rs}$.

Clustering methods are often described as one of the three following types: *hierarchial clustering, partitioning clustering and overlapping clustering.* We now give an overview of each of these methods.

I) ***Hierarchical Clustering.***

These methods study data relationships at different levels, groups are themselves grouped into bigger clusters producing in the end a tree of clusters that contains the set of all observations. The graphical representation of such a tree is called a *dendrogram.* The tree is built using either an *agglomerative* algorithm, which starts with n clusters, one observation in each, and ends with the set of all observations, or a *divisive* algorithm, which starts with one cluster containing all observations and ends with n clusters, one observation in each. We briefly describe the hierarchical methods mentioned above.

A) Agglomerative methods.

The most common agglomerative algorithms start with a single observation in each cluster and then move to $n-1$ clusters looking for:

a) **Nearest neighbour or Single linkage**. If C_1 and C_2 are clusters, the distance between clusters is defined as the smallest dissimilarity between a member of C_1 and one of C_2. At each step the clusters with the minimum distance are merged, that is

$$d_{(C_1)(C_2)} = \min\{d_{rs} : r \in C_1, s \in C_2\}.$$

b) **Furthest neighbour or Complete linkage**. The distance between clusters is defined as the largest dissimilarity between a member of C_1 and one of C_2. At each step the clusters with the minimum distance are merged, the maximum dissimilarity in this case.

$$d_{(C_1)(C_2)} = \max\{d_{rs} : r \in C_1, s \in C_2\}.$$

c) **Incremental sum of squares or Ward's method**. Clusters that minimise the increase in the total within-cluster sum of squares of the distances from the respective centroids are merged. If the increase is denoted $I_{(C_1)(C_2)}$, the distance between two clusters can be defined as

$$d_{(C_1)(C_2)} = 2I_{(C_1)(C_2)}.$$

d) The distance is defined as the average of the $n_1 n_2$ dissimilarities between all pairs.

$$d_{(C_1)(C_2)} = \sum_{r \in C_1} \sum_{s \in C_2} d_{rs}.$$

In Hartigan [37], it is also advised to bear in mind, when considering the distances between data, problems like differences in scale and correlation effects among the variables used to define the groups. Some other remarks are given about the spatial properties of agglomerative methods. Single linkage is known to be *space contracting*, that is, a cluster will move close to other clusters or individuals so that an individual will tend to add to a preexisting cluster rather than act as a new cluster center. Com-

plete linkage is *space dilating*, that is, clusters will grow so that individuals that are not yet placed in a cluster are more likely to become the nuclei of new cluster. The other agglomerative methods have spatial properties between the extremes presented by single and complete linkage.

B) Divisive methods.

The divisive methods are computationally demanding, the first step is to divide the group of n objects into two groups. It is difficult to examine all possible division, even for a small n. However, the process starts at the highest level of information and it could be stopped before it ends with n clusters containing one observation each.

II) ***Partitioning***.

The aim of these methods is to partition n objects into a specified number of non-overlapping clusters, k. The very large number of possible partitions makes it impractical to search through all the configurations for the one that optimises some criterion. Hence, partitioning methods that allow for observations to be relocated are considered.

The first step is to choose k points in a p-dimensional space as an initial partition $\mathcal{P}(n,k)$. These points or nuclei for each cluster could be chosen in a variety of ways, for example, at random, regularly spaced or mutually furthest apart, among others. The discordance between the clusters is measured by an error $e[\mathcal{P}(n,k)]$ and the optimal configuration minimises e. It is necessary to optimise by locally searching through the set of partitions to move to the next partition which has minimum $e[\mathcal{P}(n,k)]$. Observations are moved from one cluster to another. The center is updated after each addition to the cluster or after all observations have been allocated. The search ends when $e[\mathcal{P}(n,k)]$ is not reduced by a movement to the neighbourhood. Some criteria that are frequently used are the following:

a) One simple algorithm considers each observation in turn and reallocates it to the nearest centroid. After all observations are considered, the centroids are updated. The process is stopped when no objects change their group membership

b) Minimise the trace (W). Some well known criteria are based on within-cluster, W, and between-cluster, B, variation matrices. Observations are reallocated to the clustering that gives an optimum value of the criterion. Edwards and Cavalli-Sforza [21]

proposed minimising the trace$(W) = \sum_i tr(W_i)$, which is equivalent to minimising the total within-cluster sum of squares about the k centroids. There are several algorithms to find optimal sum of squares partition, for a thorough presentation see Gordon [31]. This is commonly known as the **k-means method**.

c) Other criteria have been proposed such as minimising $|W|$, minimising the trace (BW^{-1}) or density type procedures.

There are no general guidelines as which method to use in which situation but some remarks are important to make. The trace(W) criterion is simple and easily computed. It is invariant under orthogonal transformation but not under nonsingular linear transformations, hence results for raw data could differ from results for standarised data. Correlations are not taken into account in this case, therefore it tends to produce spherical clusters. Minimising $|W|$ is invariant under nonsingular linear transformations but could be influenced by a single well-clustered variable. This criteria is useful to identify well separated and same sized clusters. With the trace (BW^{-1}) criterion, if there is a partition which is very elongated in the wrong direction, the error will not be corrected in further iterations.

III) ***Overlapping clusters, clumping.***

Agglomerative algorithms that allow clusters to overlap are used for this methods, single link algorithms are generalised to k-link algorithms. However, these algorithms are not widely supported because of their computational complexity and because the resulting dendrogram is difficult to draw and interpret.

1.1.2 Model-based clustering

Several authors have considered clustering problems based on a finite mixture model. A mixture of k Gaussian distributions has frequently been used. Under the assumption that a fixed and known number of components, k, is given, the estimation problem that arises from this model consists in finding the best allocation of the n observations into the k given groups.

The use of finite mixture models to represent cluster structure is described, for example, in McLachlan and Peel [43]. In the next section we follow their presentation to give the general setting of the finite mixture models that will be used in this thesis.

1.2 Finite mixture models

The use of finite mixture models has increased considerably in the last decade and in particular they have proved useful in directly modelling population heterogeneity in a clustering context. Assuming that the observed data arise from a target population of component subpopulations in a certain proportion, then the set of clusters may be associated one to one with the components of a mixture model. In principle, fitting a finite mixture model would enable the estimation of parameters and the allocation of observations into one of the clusters. However, if there is little or no information about the number of groups present in the population it would be desirable to consider the number of components as another unknown parameter in the model.

1.2.1 General concepts

Let $\mathbf{y} = \mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n$ denote a data set, where $\mathbf{y}_j$ is a $p-$vector containing the p measurements of the analysed features for the j-th observation. Hence $\mathbf{y}_j$ is the observed value of a random variable $\mathbf{Y}_j$ with probability density function $f(\mathbf{y}_j)$ on $\mathbb{R}^p$. The density $f(\mathbf{y}_j)$ of $\mathbf{Y}_j$ is then assumed to be a k-component finite mixture density of the form

$$f(\mathbf{y}_j) = \sum_{i=1}^{k} w_i f_i(\mathbf{y}_j). \tag{1.1}$$

The mixing proportions w_i are such that,

$$0 \leq w_i \leq 1 \quad \text{and} \quad \sum_{i=1}^{k} w_i = 1. \tag{1.2}$$

If the density functions $f_i(\mathbf{y}_j)$ belong to a specified parametric family, then we can write equation (1.1) as

$$f(\mathbf{y}_j; \Psi) = \sum_{i=1}^{k} w_i f_i(\mathbf{y}_j|\boldsymbol{\theta}_i), \tag{1.3}$$

where $\Psi = (w_1, w_2, \ldots w_{k-1}, \boldsymbol{\theta})$, and $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_k)$. Notice that $w_k = 1 - \sum_{i=1}^{k-1} w_i$.

If all the density functions belong to the same parametric family, then equation (1.3) becomes

$$f(\mathbf{y}_j; \Psi) = \sum_{i=1}^{k} w_i f(\mathbf{y}_j|\boldsymbol{\theta}_i). \tag{1.4}$$

In particular, if $f(\mathbf{y}_j)$ is the density of a p-dimensional multivariate normal distribution with mean $\boldsymbol{\mu}_i$ and covariance matrix Σ_i, then the corresponding parameter vector is $\boldsymbol{\theta} = ((\boldsymbol{\mu}_1, \Sigma_1), (\boldsymbol{\mu}_2, \Sigma_2), \ldots, (\boldsymbol{\mu}_k, \Sigma_k))$ and equation (1.4) becomes

$$f(\mathbf{y}_j; \Psi) = \sum_{i=1}^{k} w_i \, N_p(\mathbf{y}_j | \boldsymbol{\mu}_i, \Sigma_i). \tag{1.5}$$

where N_p denotes a multivariate density in $\mathbb{R}^p$.

From the point of view of cluster analysis, the interest centres on identifying and interpreting this underlying mixture structure and its relationship with the sample observations.

1.2.2 Allocation latent variables

In this section we introduce the allocation vector variables as they are of crucial importance for estimation methods in finite mixture models. Consider a vector of categorical random variables Z_j for $j = 1, \ldots, n$, that take values in $1, 2, \ldots, k$. If they are regarded as allocation variables for the observations, then they are assumed to be independent draws from the distributions

$$pr(Z_j = i | \Psi) = w_i \qquad \text{for} \quad i = 1, 2, \ldots, k. \tag{1.6}$$

Conditional on the $Z_j = i$, the density of $\mathbf{Y}_j$ is given by $f_i(\mathbf{y}_j)$. If a mixture of parametric densities is being considered and the classification probabilities are of interest, they are given as

$$pr(Z_j = i | \mathbf{y}_j, \Psi) = \frac{w_i f_i(\mathbf{y}_j; \boldsymbol{\theta}_i)}{\sum_{l=1}^{k} w_l f_l(\mathbf{y}_j; \boldsymbol{\theta}_l)}, \quad \text{for} \quad i = 1, \ldots, k. \tag{1.7}$$

The vector $(z_1, z_2, \ldots, z_n)^T$ is frequently called the *missing data* part of the sample. If a sample is taken from a population with $G_1, G_2, \ldots, G_k$ groups in proportions $w_1, w_2, \ldots w_k$ and the density of $\mathbf{Y}_j$ in group G_i is given by $f_i(\mathbf{y}_j)$ then the density of $\mathbf{Y}_j$ is given by equation (1.1).

The allocation vector (z_j) plays an important role in inference. Estimation algorithms take advantage of the representation of the model through the allocation variables. A straightforward implementation of the Expectation-Maximisation (EM) algorithm, see section 1.6, uses the log *augmented-likelihood* in terms of $\mathbf{y}_j$ and Z_j to

compute the maximum likelihood estimators of the mixture distribution. In a Bayesian framework, MCMC methods are easily implemented using the full posterior conditional distribution of the missing data representation.

1.2.3 Label switching

As several authors, including McLachlan and Peel [43], emphasise, the estimation of Ψ based on the observations $\mathbf{y}_j$, is meaningful if Ψ is identifiable. In a parametric family context, this means that a distinct Ψ determines a distinct member of the family of densities $\{f(\mathbf{y}_j;\Psi) : \Psi \in \Omega\}$. Therefore,

$$f(\mathbf{y};\Psi) = f(\mathbf{y};\Psi^*) \iff \Psi = \Psi^*.$$

In a mixture model, if all the k components belong to the same parametric family, then $f(\mathbf{y};\Psi)$ is invariant under the $k!$ permutations of the component labels in Ψ. For any permutation of component indices ν of $1,\ldots,k$, the corresponding permutation of the parameter vector Ψ is given by

$$\nu(\Psi) = ((w_{\nu(1)},\ldots,w_{\nu(k)}),(\boldsymbol{\theta}_{\nu(1)},\ldots,\boldsymbol{\theta}_{\nu(k)}))$$

and the likelihood

$$L(\Psi) = \prod_{i=1}^{n}\{w_1 f(\mathbf{y}_i;\boldsymbol{\theta}_1) + \ldots + w_k f(\mathbf{y}_i;\boldsymbol{\theta}_k)\}$$

is the same for all permutations of Ψ.

The term *label switching* is used to describe the invariance of the likelihood under the relabelling of the components. This problem is often handled by imposing an artificial *identifiability constraint* on Ψ, for example ordering the mixing proportions so that $w_1 \leq w_2 \leq \ldots \leq w_k$. However, some authors such as Celeux *et al* [16], Richardson and Green [51], Stephens [60] and Frühwrith-Shnatter [27] have pointed out that this does not always give a satisfactory solution.

Richardson and Green [51] explained that different labellings of the MCMC sampler output could lead to significantly different posterior inference for the component parameters. They advised to carry out a post-process of the output according to different labellings to obtain what they called a better picture of the component parameters.

Celeux *et al* [16] used a cluster-like tool in which one of the modal regions is selected using the early iterations of the MCMC sampler, ideally before the label switching occurs, and the following sampled parameters are permuted according to the $k!$ permutation which is closer to the current cluster parameter mean.

Stephens [60] considered alternative relabelling strategies based on a decision theoretic approach to deal with label switching. For some data sets this has shown to be more efficient than the use of identifiability constraints. However, they involve an optimisation criterion, whose solution depends on the starting point selected. It was shown for several univariate examples that all optima gave qualitatively similar results. Similar results were also obtained for different choices of loss function and action space.

Frühwirth-Schnatter [27] proposed the use of a permutation sampler which, at the end of each MCMC sweep, relabels the states through a random permutation of the current labelling. The output of this permutation sampler is useful to estimate the model likelihood and to select the k number of components. For the selected number of components, the output is then used to find a suitable identifiability constraint. The model is reestimated by sampling from the constrained posterior, enabling the estimation of state specific parameters.

Dealing with the label switching problem requires a careful analysis of the particular problem. Identifiability constraints do not always throw light on the number of components in the population or the components to which observations belong. Although our main interest is not to estimate the density but to obtain information on the number of groups in the observed population and the possible classification of the observed data into those groups, label switching must be taken into account. We will describe the actions taken to deal with this problem in our context as they become necessary.

We will consider a fully Bayesian treatment of mixture models in a cluster analysis context. The analysis will be carried out through a Bayesian hierarchical model. The following section introduces the general form of Bayesian hierarchical models.

1.3 Bayesian hierarchical models

Bayesian methods are successfully used in many fields and there have been improvements in computational methods as well as theoretical results in recent years.

A thorough presentation of the foundations of Bayesian models and inference can be found in Bernardo and Smith [5]. Here we give a brief review of the general ideas required to follow the proposed method for cluster analysis.

1.3.1 Bayesian inference

As defined by Gelman *et al* [29], " Bayesian inference is the process of fitting a probability model to a set of data summarizing the result by a probability distribution on the parameters of the model and on unobserved quantities such as predictions for new observations. "

Let $Y_1, \ldots, Y_n$ be a random sample from an unknown distribution $\mathcal{F}$. Consider a parametric family of densities [1],

$$\mathcal{P} = \{p(y|\theta) : \theta \in \Theta\},$$

where the distribution $\mathcal{F}$ corresponds to one of the models in $\mathcal{P}$. The problem is then to make inference about the value of θ which corresponds to the model that best describes the data. The previous information on the unknown value of θ is described by a *prior distribution*, $p(\theta)$.

In what follows, $p(\cdot|\cdot)$ denotes a conditional probability with arguments determined by the context and $p(\cdot)$ denotes a marginal distribution. To make probability statements about a parameter θ given data y, we need a model which provides a joint probability distribution for y and θ. The joint probability mass or density function can be written as the product of the prior distribution and the *sampling or data distribution*, $p(y|\theta)$,

$$p(\theta, y) = p(\theta)p(y|\theta).$$

The posterior density can then be obtained using the *Bayes' theorem* to condition on the known value of the data y,

$$p(\theta|y) = \frac{p(\theta, y)}{p(y)} = \frac{p(\theta)p(y|\theta)}{p(y)},$$

where $p(y) = \int p(\theta)p(y|\theta)d\theta$. The latter factor does not depend on θ, with fixed y, it

[1] We concentrate on density functions but there exists an analogous formulation for discrete distribution functions.

can be considered constant leading to the unnormalised *posterior density*

$$p(\theta|y) \propto p(\theta)p(y|\theta).$$

In simple words, Bayesian methods set up a probability model, that is a joint probability distribution for all observable and unobservable quantities in a problem. Then the posterior distribution is obtained conditional on the observed data, this will be our main interest. Posterior predictive distributions $p(\tilde{y}|\mathbf{y})$ can also be obtained if they are of interest:

$$\begin{aligned} p(\tilde{y}|y) &= \int p(\tilde{y}, \theta|y)d\theta \\ &= \int p(\tilde{y}|\theta, y)p(\theta|y)d\theta \\ &= \int p(\tilde{y}|\theta)p(\theta|y)d\theta. \end{aligned}$$

Finally, the fit of the model should be evaluated as well as the sensitivity of the results to the modelling assumptions.

1.3.2 Hierarchical models

A hierarchical model is used in statistical applications which need multiple parameters to fit the data in an adequate way. Observations are modelled conditionally on parameters, which themselves have probability specifications in terms of *hyperparameters*. The joint posterior probability indicates the dependence relation among these parameters.

Consider a set of experiments or observations $y = (y_1, \ldots, y_n)$ and a parameter vector $\theta = (\theta_1, \ldots, \theta_n)$, with likelihood $p(y_j|\theta_j)$. Some of the parameters of different observations may overlap if θ_j is a vector, or coincide if θ_j is a scalar. The concept of *exchangeability* is used to create a probability model for all the parameters θ. The latter assumes that the joint probability density for the n values of θ_j, $p(\theta_1, \ldots, \theta_n)$, is invariant to permutations of the indices. This indicates that the parameters θ_j's are not distinguished from any others. A simple exchangeable distribution has each of the parameters θ_j as an independent sample from a prior distribution defined by

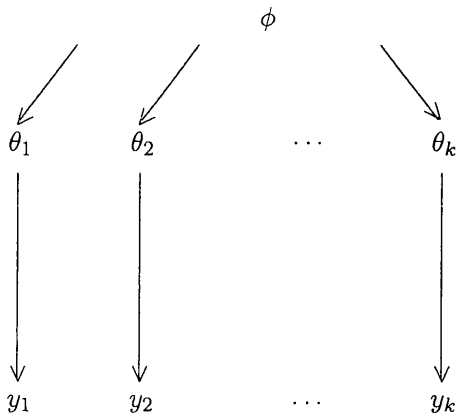


Figure 1.1: General structure of a simple hierarchical model.

parameters ϕ, given as

$$p(\theta|\phi) = \prod_{j=1}^{n} p(\theta_j|\phi).$$

Figure 1.1 shows the general structure of a simple hierarchical model, where θ_i depends on a common hyperparameter ϕ.

In general, ϕ is unknown and therefore it is given a prior distribution $p(\phi)$. The distribution for θ must average over the uncertainty in ϕ, de Finetti's theorem states that as $n \rightarrow \infty$, any suitable well-behaved exchangeable distribution on $(\theta_1, \ldots, \theta_n)$ can be written as

$$p(\theta) = \int \left[\prod_{j=1}^{n} p(\theta_j|\phi) \right] p(\phi) d\phi.$$

The basic problem in a hierarchical model is that the initial distribution of the parameters is not completely specified, it depends on the hyperparameter ϕ which has its own initial distribution. The joint prior distribution for the vector (ϕ, θ) is given by

$$p(\phi, \theta) = p(\phi)p(\theta_j|\phi),$$

and the joint posterior distribution $p(\phi, \theta|y)$ is,

$$\begin{aligned} p(\phi, \theta|y) &\propto p(\phi, \theta)p(y|\phi, \theta) \\ &= p(\phi, \theta)p(y|\theta), \end{aligned}$$

the last equality because the data distribution $p(y|\phi,\theta)$ depends only on θ.

One would attempt to give a diffuse prior to ϕ when there is not much prior information. However, care must be taken when considering an improper prior in that the posterior must be shown to be proper. Another important aspect to bear in mind is the assessment of the sensitivity of the conclusions to this simplifying assumption. Hierarchical models commonly involve a large number of parameters and numerical methods are often employed to obtain simulations form the joint posterior distribution $p(\theta,\phi|y)$. We will introduce the methods required to make inference about the parameters of the finite mixture of multivariate normal distributions, which we will assume as the underlying model for clustering, later in this chapter.

1.4 Bayesian approach to finite mixture modelling

Let us assume the number of components, k, in a mixture model is known. Let $p(\cdot)$ denote the corresponding density function for the k-component mixture. The unknown parameters $\Psi = (\mathbf{w},\boldsymbol{\theta})$ are assumed to be drawn from a set of appropriate prior distributions. Then, when considering the allocation variables, the density of the joint distribution of all variables can be written as

$$p(\mathbf{w},\mathbf{z},\boldsymbol{\theta},\mathbf{y}|k) = p(\mathbf{w}|k)p(\mathbf{z}|\mathbf{w},k)p(\boldsymbol{\theta}|\mathbf{z},\mathbf{w},k)p(\mathbf{y}|\boldsymbol{\theta},\mathbf{z},\mathbf{w},k), \tag{1.8}$$

where $p(\cdot|\cdot)$ is used to denote generic conditional distributions. Posterior quantities of interest can then be approximated by numerical methods as the EM algorithm or standard Markov chain Monte Carlo (MCMC) methods.

Suppose now that there is no prior knowledge about the number of mixture components k. As described in Green [34], using a hierarchical model in a Bayesian framework, Ψ could be extended to include k. Suppose that a joint prior for $\Psi = (k,\mathbf{w},\boldsymbol{\theta})$ is given for each k in a countable set $\mathcal{K}$. Here it is assumed that all probability densities are proper.

Imposing further conditional independence, so that $p(\boldsymbol{\theta}|\mathbf{z},\mathbf{w},k) = p(\boldsymbol{\theta}|k)$ and $p(\mathbf{y}|\boldsymbol{\theta},\mathbf{z},\mathbf{w},k) = p(\mathbf{y}|\boldsymbol{\theta},\mathbf{z})$, then

$$p(k,\mathbf{w},\mathbf{z},\boldsymbol{\theta},\mathbf{y}) = p(k)p(\mathbf{w}|k)p(\mathbf{z}|\mathbf{w},k)p(\boldsymbol{\theta}|k)p(\mathbf{y}|\boldsymbol{\theta},\mathbf{z}). \tag{1.9}$$

Several alternatives to sample from $p(k, \mathbf{w}, \mathbf{z}, \boldsymbol{\theta}, \mathbf{y})$ are given in recent literature. These methods are an extension of MCMC methods known as trans-dimensional MCMC, Green [32], [33], [34]. In this thesis we will describe their performance when attempting to approach the problem of multivariate model-based cluster analysis with an unknown number of components. The simulation methods we will use in throughout this work are based on Markov chain Monte Carlo methods, a review of the general concepts will be given in the following sections.

1.5 Markov chain Monte Carlo samplers

In the Bayesian framework described in the previous sections, the interest centres on the posterior distribution of all variables (parameters and allocation variables) given data $\mathbf{Y}$. Let π be the objective distribution, commonly referred to as the *target distribution.*

In straightforward cases π is a probability distribution with respect to some measure, usually Lebesgue or counting measure. The use of Markov chain Monte Carlo (MCMC) samplers comes into practice when directly generating samples from π is not possible. Simulation methods are nowadays well known and widely used, see Robert and Casella [53], Liu [40], Gamerman [28], for a detailed presentation. Some properties of Markov chains are essential for the study of MCMC methods, we present a summary in the next section.

1.5.1 Some properties of a Markov chain

A *Markov chain* is a sequence of random variables where the value X_{t+1} depends on the history of the chain only through the value of X_t. Following Robert and Casella [53], a definition of a Markov chain will be given in terms of its transition kernel, that is, the function that determines the transition probabilities.

Definition 1.1. *A **transition kernel** is a function P defined on $\mathcal{X} \times \mathcal{B}(\mathcal{X})$ such that*

i) $\forall x \in \mathcal{X}, P(x, \cdot)$ *is a probability measure.*

ii) $\forall A \in \mathcal{B}(\mathcal{X}), P(\cdot, A)$ *is measurable.*

where $\mathcal{B}(\mathcal{X})$ denote the Borel sets on $\mathcal{X}$.

In the discrete case, the transition kernel is simply a transition matrix $P = \{P_{xy}\}_{x,y\in\mathcal{X}}$ with transition probabilities

$$P_{xy} = P(X_n = y|X_{n-1} = x), \quad x, y \in \mathcal{X}.$$

Definition 1.2. *Given a transition kernel P, a sequence $X_0, X_1, \ldots X_n, \ldots$ of random variables is a* ***Markov chain,*** *denoted (X_n) , if for any t the conditional distribution of X_t given $X_{t-1}, X_{t-2}, \ldots, X_0$ is the same as the distribution of X_t given X_{t-1},*

$$P(X_{t+1} \in A|X_0 = x_0, X_1 = x_1, \ldots, X_t = x_t) = P(X_{t+1} \in A|X_t = x_t) = \int_A P(x_t, dx).$$

The chain is *time homogeneous* if the probability $P(X_{t+1} \in A|X_t = x_t)$ does not change with t. (X_n) is usually defined for the index $n \in \mathbb{N}$, in our context, it will mostly represent the iterations of a sampling scheme. It is of interest to compute the n-step transition probability defined as $P^n(x_t, A) = P(X_{t+n} \in A|X_t = x_t)$ which can be computed from the *Chapman-Kolmogorov* equations

$$P^{n+m}(x, A) = \int_{\mathcal{X}} P^n(y, A)P^m(x, dy),$$

where $P^1(x, A) = P(x, A)$.

An important property enjoyed by all the Markov chains generated through a MCMC procedure is the *stationarity* property. In general, for any Markov process, this property translates into the existence of an invariant probability measure.

Definition 1.3. *A σ-finite measure π is invariant for the transition kernel $P(\cdot, \cdot)$ if*

$$\pi(B) = \int_{\mathcal{X}} P(x, B)\pi(dx), \forall B \in \mathcal{B}(\mathcal{X}).$$

In particular, when $\pi(\mathcal{X}) < \infty$, then the normalized version is known as the invariant or stationary distribution.

One of the properties required for a stationary distribution to exist is that the kernel P allows for free moves all over the state space, this property is known as irreducibility.

Definition 1.4. *Given a measure φ, the Markov chain (X_n) with transition kernel $P(x, y)$ is φ-irreducible if, for every $A \in \mathcal{B}(\mathcal{X})$, with $\varphi(A) > 0$, there exists an $n \in \mathbb{N}$ such that $P^n(x, A) > 0$ for all $x \in \mathcal{X}$.*

In words, a state $x \in \mathcal{X}$ is said to be irreducible if under the transition rule one has nonzero probability of moving from x to any other state and then coming back in a finite number of steps.

A Markov chain is *aperiodic* if the maximum common divider of the number of steps it takes for the chain to come back to any starting point is equal to one. In an aperiodic chain, there are no portions of the state space which the chain can only visit at certain regularly spaced times. If a chain has a proper invariant distribution π and it is irreducible and aperiodic, then π is the unique invariant distribution and also the equilibrium distribution of the chain (Numelin [46], Tierney [63]).

An MCMC sampler produces an aperiodic and irreducible Markov chain of random variables which satisfies the detailed balance condition so that its stationary distribution is the corresponding target distribution π. The detailed balance condition below will ensure the general balance given in Definition 1.2.

Definition 1.5. *The detailed balance equation is given by*

$$\pi(x)P(x, dx') = \pi(x')P(x', dx) \tag{1.10}$$

for every $x, x' \in \mathcal{X}$.

From the definition above we can see that

$$\begin{aligned}
\int_{\mathcal{X}} \pi(dx)P(x, B) &= \int_{\mathcal{X}} \pi(dx) \int_B P(x, dx') \\
&= \int_{\mathcal{X}} \int_B \pi(dx)P(x, dx') \\
&= \int_{\mathcal{X}} \int_B \pi(dx')P(x', dx) \\
&= \int_{\mathcal{X}} \pi(dx') \int_B P(x', dx) \\
&= \pi(B).
\end{aligned}$$

1.5.2 Metropolis-Hastings methods

The Metropolis algorithm was introduced by Metropolis *et al* [45] and a generalization was given by Hastings [38], it can be used to generate samples from a target distribution π known up to a normalising constant.

Following Green [34], let us consider a Metropolis-Hastings in a general state space. In order to sample from the target distribution π of a random quantity x on a state

space $\mathcal{X}$, in the construction of the desired Markov chain, only reversible chains are considered. Therefore, the transition kernel P satisfies the detailed balance condition

$$\int_{(x,x')\in A\times B} \pi(dx)P(x,dx') = \int_{(x,x')\in A\times B} \pi(dx')P(x',dx), \tag{1.11}$$

for all Borel sets $A, B \in \mathcal{X}$.

In the Metropolis-Hastings sampler, a transition is made by drawing a candidate value x' from an arbitrary proposal distribution $q(x,dx')$ whose support includes $\mathcal{X}$ and accepting the proposed move with probability $\alpha(x,x')$. If the new state is rejected the sampler stays in the current state, then $P(x,dx')$ has an atom at x. This would contribute the same quantity to both sides of equation (1.11), namely $\int_{A\bigcap B} P(x,\{x\})\pi(dx)$. Subtracting this quantity leaves

$$\int_{(x,x')\in A\times B} \pi(dx)q(x,dx')\alpha(x,x') = \int_{(x,x')\in A\times B} \pi(dx')q(x',dx)\alpha(x',x). \tag{1.12}$$

In Green [32] it is shown that $\pi(dx)q(x,dx')$ is dominated by a symmetric measure μ on $\mathcal{X}\times\mathcal{X}$. Let f be its density with respect to μ, then equation (1.12) becomes

$$\int_{(x,x')\in A\times B} \alpha(x,x')f(x,x')\mu(dx,dx') = \int_{(x,x')\in A\times B} \alpha(x',x)f(x',x)\mu(dx',dx). \tag{1.13}$$

Since μ is a symmetric measure, the latter is satisfied for all Borel sets A, B if

$$\alpha(x,x') = min\left\{1, \frac{f(x,x')}{f(x',x)}\right\},$$

or commonly written

$$\alpha(x,x') = min\left\{1, \frac{\pi(dx')q(x',dx)}{\pi(dx)q(x,dx')}\right\}. \tag{1.14}$$

The Metropolis-Hastings has by construction an invariant distribution π, Tierney [63] showed that Metropolis algorithm is almost surely aperiodic, hence the chain produced will become stationary at its invariant distribution π. If the chain is run long enough, the samples produced by the chain can be regarded as samples from the target distribution π.

1.5.3 Gibbs sampler method

The Gibbs sampler is an MCMC method where the transition kernel is formed by the full conditional distributions. Suppose that for some $p > 1$, the random variable $X \in \mathcal{X}$ with distribution π, can be decomposed as $X = (X_1, \ldots, X_p)$. Suppose that the corresponding univariate conditional densities $f_1, \ldots, f_p$ are known and can be sampled from, namely

$$X_i | x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_p \sim f_i(X_i | x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_p)$$

for $i = 1, \ldots, p$.

The Gibbs sampler has the following one-step ahead transition from $X^{(t)}$ to $X^{(t+1)}$, given
$x^{(t)} = (x_1^{(t)}, \ldots, x_p^{(t)})$,

$$\begin{aligned}
X_1^{(t+1)} &\sim f_1(x_1 | x_2^{(t)}, \ldots, x_p^{(t)}), \\
X_2^{(t+1)} &\sim f_1(x_2 | x_1^{(t+1)}, x_3^{(t)} \ldots, x_p^{(t)}), \\
&\vdots \\
X_p^{(t+1)} &\sim f_1(x_p | x_2^{(t+1)}, \ldots, x_{p-1}^{(t+1)}).
\end{aligned}$$

In the above description a fixed updating order for the components of X_j is assumed, although this is common, it is not necessary. Random permutations of the updating order are acceptable. Moreover, not all components need to be updated at each iteration.

The Gibbs sampler is a special case of the Metropolis-Hastings algorithm with the particular feature that the acceptance rate is uniformly equal to 1, therefore every simulated value is accepted. Tierney [63] establishes formal convergence conditions for the Gibbs sampler. Despite the theoretical results ensuring the convergence of the Gibbs sampler, its practical implementation may be complicated because of the complexity of the models considered.

1.6 Other estimation methods

Inference for the posterior distribution of a Bayesian model for a finite mixture of distributions with a given number of components, k, has also been performed through the maximum likelihood (ML) estimation of the unknown parameters.

The *Expectation Maximisation* (EM) algorithm was used by Dempster *et al* [20] to overcome difficulties in maximising likelihoods by taking advantage of the missing data representation of the model. Following McLachlan and Peel [43], we briefly present the application of the EM algorithm for the ML fitting of the parametric mixture model given by equation (1.4), where we consider $\Psi = (w_1, \ldots, w_{k-1}, \xi) \in \Omega$ and $\xi = (\theta_1, \ldots, \theta_k)$.

The log likelihood for ψ formed from the observed data $\mathbf{y} = (\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n)$ is given by

$$\begin{aligned} \log(L(\psi)) &= \sum_{j=1}^{n} \log f(\mathbf{y}_j|\psi) \\ &= \sum_{j=1}^{n} \log \left\{ \sum_{i=1}^{k} w_i f_i(\mathbf{y}_j|\theta_i) \right\}. \end{aligned} \tag{1.15}$$

The computation of the MLE of ψ commonly requires solving the likelihood equation

$$\frac{\partial \log L(\psi)}{\partial \psi} = 0,$$

which is not always an easy task.

Consider now the missing data formulation described in section 1.2.2, here we will consider the vector $z_{ij} = (z_j)_i = 1$ or 0, for $i = 1, \ldots, k$ and $j = 1, \ldots, n$, according to whether $\mathbf{y}_j$ did or did not arise from the i-th component of the mixture. The log augmented-likelihood given by equation (1.15) could be written as

$$\log(L(\psi)) = \sum_{i=1}^{k} \sum_{j=1}^{n} z_{ij} \left\{ \log w_i + \log f_i(\mathbf{y}_j|\theta_i) \right\}. \tag{1.16}$$

The EM algorithm proceeds iteratively in the expectation step (E) and the maximisation step (M). Let $\psi^{(0)}$ be the initial value for ψ, the E-step for the $(g+1)$-th

iteration, requires the computation of the conditional expectation

$$Q(\psi;\psi^{(g)}) = \mathbf{E}_{\psi^{(g)}}\{\log L(\psi)|\mathbf{y}\}. \tag{1.17}$$

Since the log likelihood given by equation (1.16) is linear in the unobservable data z_{ij}, the E-step simply requires the calculation of the current conditional expectation of Z_{ij} given the observed data $\mathbf{y}$, where Z_{ij} is the random variable corresponding to z_{ij}. Now,

$$\mathbf{E}_{\psi^{(g)}}(Z_{ij}|\mathbf{y}) = \frac{w_i^{(g)} f_i(\mathbf{y}_j;\boldsymbol{\theta}_i^{(g)})}{\sum_{l=1}^{k} w_l^{(g)} f_l(\mathbf{y}_j;\boldsymbol{\theta}_l^{(g)})},$$

using this we have that on taking the conditional expectation of (1.16)

$$Q(\psi;\psi^{(g)}) = \sum_{i=1}^{k}\sum_{j=1}^{n}\left\{\frac{w_i^{(g)} f_i(\mathbf{y}_j;\boldsymbol{\theta}_i^{(g)})}{\sum_{l=1}^{k} w_l^{(g)} f_l(\mathbf{y}_j;\boldsymbol{\theta}_l^{(g)})}\right\}\{\log w_i + \log f_i(\mathbf{y}_j|\theta_i)\}.$$

The M-step on the $(g+1)$-th iteration requires the global maximisation of $Q(\psi;\psi^{(g)})$ with respect to ψ over the parameter space Ω to give the updated estimate $\psi(g+1)$. The updated estimates of the mixing proportions w_i for the $(g+1)$-th iteration are calculated independently of the updated estimates of the parameter vector ξ containing the unknown parameters in the component densities.

The update estimate of w_i is given as

$$w_i^{(g+1)} = \frac{1}{n}\sum_{j=1}^{n}\frac{w_i^{(g)} f_i(\mathbf{y}_j;\boldsymbol{\theta}_i^{(g)})}{\sum_{l=1}^{k} w_l^{(g)} f_l(\mathbf{y}_j;\boldsymbol{\theta}_l^{(g)})}, \tag{1.18}$$

for $i = 1,\ldots,k$. There is a contribution from each observation $\mathbf{y}_j$ to form the estimate of w_i on the $(g+1)$ iteration which is equal to its posterior probability of membership of the i-th component of the mixture model.

The update of ξ on the $(g+1)$-th iteration is obtained as the appropriate root of

$$\sum_{i=1}^{k}\sum_{j=1}^{n}\left\{\frac{w_i^{(g)} f_i(\mathbf{y}_j;\boldsymbol{\theta}_i^{(g)})}{\sum_{l=1}^{k} w_l^{(g)} f_l(\mathbf{y}_j;\boldsymbol{\theta}_l^{(g)})}\right\}\{\partial \log f_i(\mathbf{y}_j|\theta_i)/\partial\xi\} = 0.$$

The solution to the latter expression often exists in a closed form. The E and M-steps are alternated until the difference $L(\psi^{(k+1)}) - L(\psi^{(k)})$ changes by an arbitrarily small amount in the case of the convergence of the sequence of values $\{L(\psi^{(k)})\}$.

A detailed account of convergence properties of the EM algorithm is given in Wu [67]. In practice, graphical or multiple starting values are used to check that a maximum has been reached. The output of the E-M algorithm would limit the inference to the resulting estimates.

1.7 Trans-dimensional models

The estimation of a mixture model with an unknown number of components involves a change in the dimension of the parameter space and the use of standard MCMC samplers is not possible. However, a direct approach to compute the joint posterior $p(\psi_k|\mathbf{y})$, given by equation (1.9), via MCMC, sometimes called *across-model* simulation (Green [34]), was introduced as the *reversible jump* MCMC in Green [32]. The use of the Metropolis-Hastings paradigm to build a suitable reversible chain is briefly presented.

1.7.1 Reversible jump Markov chain Monte Carlo

From the construction of the Metropolis-Hastings sampler given in the previous section, the present section describes the so called *trans-dimensional* Metropolis-Hastings known as the reversible jump MCMC sampler.

Let x be the state of a random variable where $\mathcal{X} \subset \mathbb{R}^p$ and suppose π has a density with respect to p-dimensional Lebesgue measure. Then consider a family of "moves" indexed by $m = 1, 2 \ldots$, including those between dimensions.

Suppose the move m from x to x', a higher dimensional space is attempted. In most practical applications this method requires that, at the current state x, r random numbers u are generated from a known density h. Here r is the difference in the number of dimensions between x and x', then the so called *dimension matching* is satisfied, namely, $p + r = p' + r'$. Once this is done, a new state is formed using a suitable deterministic function $l : \mathbb{R}^p \times \mathbb{R}^r \to \mathbb{R}^{p'}$, of the current state and the random numbers so that the new state $x' = l(x, u)$. To reverse the transition form x' to x the random numbers $u' \sim h$ are used so that $x = l'(x', u')$, where $l' : \mathbb{R}^{p'} \times \mathbb{R}^{r'} \to \mathbb{R}^p$. In practice, defining the above functions is not a straightforward task, finding good candidates can be difficult. If the transformation from (x, u) to (x', u') is a bijection, and it is possible to differentiate both the inverse of the transformation and the transformation itself,

the detailed balance condition is given by

$$\int_{(x,x')\in A\times B} \pi(dx)q_m(x,dx')\alpha_m(x,x') = \int_{(x,x')\in A\times B} \pi(dx')q_m(x',dx)\alpha_m(x',x). \quad (1.19)$$

for all m, A, B. Here $q_m(x,dx')$ is the joint distribution of move type m and destination x'. To obtain the complete transition kernel, one must sum over m, so that for $x \notin B$, $P(x,B) = \sum_m \int_B q_m(x,dx')\alpha_m(x,x')$.

Then it can be seen that the move is accepted with probability

$$\alpha_m(x,x') = \min\left\{1, \frac{\pi(x')r_m(x')h'_m(u')}{\pi(x)r_m(x)h_m(u)}\left|\frac{\partial(x',u')}{\partial(x,u)}\right|\right\}, \quad (1.20)$$

where $r_m(x)$ is the probability of choosing the move type m when in state x, and $h_m(u)$ is the density function of u. The final term is taking into account the Jacobian for the transformation from (x,u) to (x',u'). If the dimension matching failed, the mapping and its inverse could not both be differentiable.

1.7.2 Birth and death Markov chain Monte Carlo

An alternative method to make inference about a finite mixture model with an unknown number of components was proposed by Stephens [59]. This method consists of the construction of a continuous time Markov birth-death process with the appropriate stationary distribution. The parameters of the model are viewed as a (marked) point process, with each point representing a component of the mixture. In our context, the points represent the mean-covariance pairs and are marked by the component weights. The number of components is allowed to vary by continuous births and deaths with relative rates determining the stationary distribution of the process.

Markov birth and death process

We are interested in inference for the parameters of a point process model that corresponds to the hierarchical Bayesian model for a mixture of normal distributions as given by equation (1.9). Following Stephens [59] we will describe the methodology.

Assume that the pairs $(w_1,\theta_1),\ldots,(w_k,\theta_k)$ are exchangeable. Suppose that the prior distribution for $(k,\mathbf{w},\boldsymbol{\theta})$ given the corresponding hyperparameters, denoted by ω, is of the form

$$r(k,\mathbf{w},\boldsymbol{\theta}|\omega) = p(k|\omega)p(\mathbf{w},\boldsymbol{\theta}|k,\omega) \quad (1.21)$$

where k has a prior probability mass distribution $p(k|\omega)$. Suppose $\boldsymbol{\theta}$ and $\mathbf{w}$ are a priori independent given k, with $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_k$ being independent and identically distributed on a space Θ from a distribution with density $\tilde{p}(\boldsymbol{\theta}|\omega)$ and $\mathbf{w}$ having a Dirichlet distribution

$$p(\mathbf{w}|k, \omega) \sim D(1, \ldots, 1).$$

With this simple case of a prior distribution for $(k, \mathbf{w}, \boldsymbol{\theta})$, exchangeability is ensured since given k, $r(\cdot)$ is invariant under relabelling of the components, in that

$$r(k, (w_1, \ldots, w_k), (\theta_1, \ldots, \theta_k)) = r(k, (w_{\nu(1)}, \ldots, w_{\nu(k)}), (\theta_{\nu(1)}, \ldots, \theta_{\nu(k)})),$$

for all permutations ν of $1, \ldots, k$.

A description of the construction of an irreducible Markov chain with stationary distribution $p(k, \mathbf{w}, \boldsymbol{\theta}|\mathbf{y}, \omega)$ will be described in the following section.

Construction of a Markov chain via simulation of point processes

Since the prior distribution of $(\mathbf{w}, \boldsymbol{\theta})$ does not depend on the labelling of the components, and the likelihood $L(k, \mathbf{w}, \boldsymbol{\theta})$ is also invariant under permutations of the component labels, the posterior distribution

$$p(k, \mathbf{w}, \boldsymbol{\theta}|\mathbf{y}, \omega) \propto L(\mathbf{y}|k, \mathbf{w}, \boldsymbol{\theta}) p(k, \mathbf{w}, \boldsymbol{\theta}|\omega) \tag{1.22}$$

will also be invariant. Therefore, the labelling of the components can be ignored and any set of k parameter values $\{(w_1, \boldsymbol{\theta}_1), \ldots, (w_k, \boldsymbol{\theta}_k)\}$ can be seen as a set of k points in $[0, 1] \times \Theta$ where $\sum_{i=1}^{k} w_i = 1$.

The posterior distribution $p(k, \mathbf{w}, \boldsymbol{\theta}|\mathbf{y}, \omega)$ is then seen as a marked point process on $[0, 1] \times \Theta$, with each $\boldsymbol{\theta}_i$ associated with a mark $w_i \in [0, 1]$, these marks being constrained to add up to one.

To simulate the point process, Stephens [59], following Ripley [52], constructs a continuous time Markov process with $p(k, \mathbf{w}, \boldsymbol{\theta}|\mathbf{y}, \omega)$ as stationary distribution keeping ω fixed. This process is combined with standard MCMC update steps to create a Markov chain with stationary distribution $p(k, \mathbf{w}, \boldsymbol{\theta}, \omega|\mathbf{y})$.

A birth and death process for the components of a mixture model is constructed on the state space $\Omega = \bigcup_{k \geq 1} \Omega_k$, where Ω_k is the parameter space of the mixture

model with k components, ignoring the labelling. A member of Ω_k will be denoted $y = \{(w_1, \boldsymbol{\theta}_1), \ldots, (w_k, \boldsymbol{\theta}_k)\}$. Births and deaths are restricted to be of the following form: when the process is at $y \in \Omega_k$ at time t,

- ◇ A birth is said to occur at $(w, \boldsymbol{\theta}) \in [0,1] \times \Theta$, the process jumps to a new state

$$y \cup (w, \boldsymbol{\theta}) := \{(w_1(1-w), \boldsymbol{\theta}_1), \ldots, (w_k(1-w), \boldsymbol{\theta}_k), (w, \boldsymbol{\theta})\} \in \Omega_{k+1}.$$

- ◇ A death is said to occur at $(w_i, \boldsymbol{\theta}_i) \in y$, the process jumps to a new state

$$y \setminus (w_i, \boldsymbol{\theta}_i) := \{(\frac{w_1}{(1-w_i)}, \boldsymbol{\theta}_1), \ldots, (\frac{w_k}{(1-w_i)}, \boldsymbol{\theta}_k)\} \in \Omega_{k-1}.$$

In this way a birth increases the number of components by one and a death decreases the number of components by one. Note that the constraint $\sum_{i=1}^{k} w_i = 1$ is satisfied after a birth or a death. When the process is at $y \in \Omega_k$, births and deaths occur as independent Poisson processes as follows:

- ◇ **Births**: These occur with overall rate $\beta(y)$ and when a birth occurs, it occurs at point $(w, \boldsymbol{\theta}) \in [0,1] \times \Theta$ chosen according to density $b(y; (w, \boldsymbol{\theta}))$.

- ◇ **Deaths**: When the process is at $= \{(w_1, \boldsymbol{\theta}_1), \ldots, (w_k, \boldsymbol{\theta}_k)\} \in \Omega_k$ each point (w_j, ϕ_j) dies independently of others as a Poisson process with rate

$$\delta_j = d(y \setminus (w_j, \boldsymbol{\theta}_j); (w_j, \boldsymbol{\theta}_j)),$$

  for some $d : \Omega \times ([0,1] \times \Theta) \to \mathbb{R}^+$, the overall death rate is given by $\delta(y) = \sum_j \delta_j(y)$.

The time to the next birth/death event is then exponentially distributed with mean $1/(\beta(y) + \delta(y))$ and it will be a birth with probability

$$Pr(birth) = \frac{\beta(y)}{(\beta(y) + \delta(y))}$$

and a death of component j with probability

$$Pr(death) = \frac{\delta_j(y)}{(\beta(y) + \delta(y))}.$$

Stephens [59] showed that, assuming the general hierarchical prior on $(k, w, \boldsymbol{\theta})$ given by $r(\cdot)$ and keeping ω fixed, the birth and death process defined above has stationary distribution $p(k, w, \boldsymbol{\theta}|x^n, \omega, n)$, provided b and d satisfy the following condition

$$(k+1)d(y;(w,\boldsymbol{\theta}))r(y \cup (w,\boldsymbol{\theta}))L(y \cup (w,\boldsymbol{\theta}))k(1-w)^{k-1} = \beta(y)b(y;(w,\boldsymbol{\theta}))r(y)L(y)$$

$\forall y \in \Omega_k$ and $(w, \boldsymbol{\theta}) \in [0,1] \times \Theta$. Here $L(y)$ is the likelihood in state y.

He considers the special case

$$r(y) = p(k|\omega)\tilde{p}(\boldsymbol{\theta}_1|\omega)\tilde{p}(\boldsymbol{\theta}_2|\omega)\cdots\tilde{p}(\boldsymbol{\theta}_k|\omega),$$

where one can sample from $\tilde{p}(\theta_i|\omega)$ and

$$\begin{aligned} \beta(y) &= \lambda_b, \quad \text{a constant,} \\ b(y,(w,\boldsymbol{\theta})) &= k(1-w)^{k-1}\tilde{p}(\boldsymbol{\theta}|\omega). \end{aligned}$$

This process has the desired stationary distribution provided that, when the process is at $y = \{(w_1, \boldsymbol{\theta}_1), \ldots, (w_k, \boldsymbol{\theta}_k)\}$, each point $(w_j, \boldsymbol{\theta}_j)$ dies independently of the others as a Poisson process with a rate

$$d(y \setminus (w_j, \boldsymbol{\theta}_j); (w_j, \boldsymbol{\theta}_j)) = \lambda_b \frac{L(y \setminus (w_j, \boldsymbol{\theta}_j))}{L(y)} \frac{p(k-1|\omega)}{kp(k|\omega)}. \tag{1.23}$$

Cappé *et al* [12] look at the similarity between the reversible jump and the birth-and-death sampling methods. They show that for any BDMCMC process satisfying some weak regularity conditions, a sequence of RJMCMC processes can be constructed so that the it converges to the BDMCMC process on appropriate rescaling of the time. For a rigorous proof see Cappé *et al* [12]. In broad terms, for $N \in \mathbb{N}$ they define a RJMCMC sampler with birth-and-death probabilities

$$\begin{aligned} b_N(\theta) &= 1 - exp\{-\beta(\theta)/N\}, \\ d_N(\theta) = 1 &- b_N(\theta) = exp\{-\beta(\theta)/N\}, \end{aligned}$$

where $\beta(\theta)$ is the birth-rate of the BDMCMC. The acceptance probability for the RJMCMC depends on N. For each N they construct a continuous process $\{\theta^N_{(t)}\}$ as $\theta^N_{(t)} = \theta^N_{\lfloor Nt \rfloor}$, where $\lfloor \cdot \rfloor$ denotes the integer part. The state of the BDMCMC sampler

at time $t \geq 0$ is denoted by $\theta_{(t)}$. They show that, as $N \to \infty$, a birth is rarely proposed but always accepted and a death is almost always proposed but rarely accepted. Both schemes result in waiting times that are asymptotically exponentially distributed with rates in accordance with the BDMCMC sampler. As $N \to \infty$, the process $\{\theta^N_{(t)}\}$ and $\{\theta_{(t)}\}$ become increasingly similar.

Cappé *et al* [12] also show that discrete moves, such as split/combine moves, can be included in continuous time MCMC samples. However, they conclude that these moves do not significantly improve the accuracy of the results. This agrees with the fact that the split/combine moves require to be carefully tuned to obtain good acceptance probabilities. Although BDMCMC method is related to RJMCMC, in that convergence of the reversible jump chain to a limiting continuous time birth-and-death chain can been shown, the methodology itself is different but both algorithms present a relatively similar complexity. The fact that continuous samplers are able to move to unlikely places could be considered an advantage.

1.8 Thesis outline

The outline of the following chapters of this thesis is as follows: Chapter 2 is dedicated to the literature review of cluster analysis. Some papers on cluster analysis are briefly described, we concentrate on model-based cluster analysis, although some other approaches relevant to the development of the thesis will be included.

In Chapter 3 we will introduce the use of the RJMCMC sampler to estimate densities of a univariate mixture of normal distributions. Next, our extension for the mixture of multivariate normal distributions using a moment matching type split/combine move and a birth/death move, will be given. A sensitivity analysis for posterior inference will also be included in this chapter. Following the results obtained in Chapter 3, Chapter 4 will discuss the use of a data informed split/combine moves in the RJMCMC methodology. We describe the results in terms of efficiency of the sampler and description of the groups.

In Chapter 5 we consider the use of the BDMCMC sampler to carry out model-based cluster analysis in high dimensional data. In general, the results for the cluster analysis give an adequate description of the data for well separated clusters. We explore the use of a data driven prior distribution for the mean vectors to increase the efficiency

of the sampler.

The convergence for a trans-dimensional sampler is not easily assessed. We will confine Chapter 6 to present the use of some techniques recently proposed in literature to evaluate the convergence of both the RJMCMC and BDMCMC samplers.

We have found that in practical clustering, it is often difficult to describe the behavior of data that belong to the same group with a single multivariate normal distribution, or in fact any multivariate distribution. The efficiency and in some sense the attractive simplicity of the Gaussian mixture model is not always the best option to capture the structure of the data. In Chapter 7 we consider representing a complex cluster structure as a mixture of standard normal components. We consider two models based on mixtures of restricted normal distributions for each component. The first model has a covariance matrix given as $\tau_j^{-1} I_p$, where $\tau_j > 0$ and I_p denotes a $p \times p$ identity matrix, for each component. The second model considers a diagonal covariance matrix $\mathcal{D}_j = (\tau_{j_1}^{-1}, \ldots, \tau_{j_p}^{-1})$ for each component. We place a tight restriction on the covariance matrices through the prior distributions given to parameters τ_j and τ_{jl} and discuss criteria to define submixtures of the resulting fitted mixture which describe one group.

Finally, Chapter 8 will include the general conclusions and describe some aspects that have been raised throughout the thesis that could lead to future work.

CHAPTER 2

Literature review

Probability models have been used as basis for cluster analysis for quite some time and the literature is extensive. In this chapter, we present a general review of some of the work that has been done in this area. Initially, it was the practice to restrict the distributions of the components of the mixture models in order to make estimation possible as sufficient computational power was not available. Later on, less restrictions were imposed and a larger variety of mixture models was explored. We will concentrate throughout this work on a Bayesian approach to the problem. Mixtures of distributions have also been treated using Bayesian semi-parametric models based on Dirichlet processes mixture models.

We present other approaches to clustering which are based on low dimensional projections of the data and graphical tools. Ideas from these works are related to some moves proposed in this thesis for a RJMCMC sampler. Finally, an important aspect in clustering is deciding how many clusters to consider. We briefly present some of the recent associated literature.

2.1 Gaussian mixtures for cluster analysis

Several authors have considered a finite mixture of Gaussian distributions as the underlying statistical model for a cluster analysis. This on one hand led to the development of new clustering methods and on the other it brought an insight into when a particular clustering method might be expected to work well. It has been shown that some widely used heuristic methods correspond to approximate estimation methods for a certain probability model.

When the number of components, k, has been specified, the clustering problem centers on the estimation of the parameters of mixture components from which each observation originated. Symons [61] presented results for both maximum likelihood and Bayesian approaches. He considered the mixture components to have both equal and not necessarily equal covariance matrices. His solutions find the allocation which minimises criteria which are modifications to the determinant of the within groups sum of squares criterion. In the examples he explored, he noted that local minima were found for many data sets.

Since high speed computers became available, the EM algorithm has been the most common approach to fit finite mixtures models through ML estimators. In particular, clustering with a finite mixture of multivariate normal distributions as the underlying statistical model requires a maximum likelihood approach. Celeux and Govaert [15] divide the maximum likelihood approaches taken in this context into the mixture approach and the classification approach. In the mixture approach, the parameter Ψ is chosen to maximise the loglikelihood given by equation (1.15). Solutions to this problem can be found through the EM algorithm as presented in section 1.6.

The classification approach uses the classification likelihood given by

$$p(y|\boldsymbol{\theta}, z) = \prod_{i=1}^{n} w_i \, N_p(\mathbf{y}_j|\boldsymbol{\mu}_{z_i}, \Sigma_{z_i}). \tag{2.1}$$

The computation of the classification maximum likelihood (CML) using the decomposition of the covariance matrices given by (2.2) is done through a version of the EM algorithm known as the CEM algorithm. An iteration of the CEM consists in computing the conditional probabilities for the allocation parameter $p(z_i = k|\cdots)$; then an updated partition is calculated by assigning each $\mathbf{y}_i$ to the cluster which provides the maximum current conditional probability for the allocation; and the ML estimates $\hat{\boldsymbol{\theta}}$ are computed using this partition.

Celeux and Govaert [15] considered fourteen different models for the covariance structure, based on the eigenvalue decomposition, for bidimensional data. They focused on the most parsimonious models that allow for a different volume among components. After numerical experiments they concluded that these are capable of describing many clustering structures without needing complex algorithms. They recommend the use

of the following forms of the covariance matrices:

$$\begin{aligned}
\Sigma_k &= \sigma_k^2 I, \quad \sigma_k^2 \text{ unknown}, 1 \leq k \leq K; \\
\Sigma_k &= \lambda_k \text{Diag}(\sigma_1^2, \ldots, \sigma_p^2), \quad \lambda_1, \ldots, \lambda_k, \sigma_1^2, \ldots, \sigma_k^2 \text{ unknown}; \\
\Sigma_k &= \lambda_k \Sigma, \quad \Sigma \text{ unknown and symmetric.}
\end{aligned}$$

2.2 Bayesian clustering with restricted covariance structure

Other authors considered the estimation problem from a Bayesian approach. Banfield and Raftery [3] carried out cluster analysis based on a mixture of multivariate normal distributions as given by equation (1.8). The covariance matrices are decomposed as

$$\Sigma_k = \lambda_k D_k A_k D_k', \tag{2.2}$$

where λ_k is a scalar that controls the volume of the k-th group, $A_k = \text{diag}\{1, a_{k2}, \ldots, a_{kp}\}$, for $1 \geq a_{k2}, \geq \cdots \geq a_{kp}$, its shape and D_k, an orthogonal matrix, its orientation.

They developed algorithms to maximise the classification likelihood given by equation (2.1). The method is implemented in the software MCLUST, as both an S-PLUS function and a Fortran program available from StatLib, the latter can be obtained at htpp://lib.stat.cmu.edu/general/mclust. However, the algorithm has some limitations. It assumes that the mixing proportions w_i are equal, estimates for the model parameters $\boldsymbol{\theta} = ((\boldsymbol{\mu}_1, \Sigma_1), (\boldsymbol{\mu}_2, \Sigma_2), \ldots, (\boldsymbol{\mu}_k, \Sigma_k))$ tend to be biased. The number of components, k, is chosen via a statistic based on an approximation of posterior probabilities $p(k = r|\mathbf{y})$, called the Approximate Weight of Evidence (AWE). It is a heuristically derived approximation to twice the log of the Bayes factor. This statistic can be used to compare results obtained with different values of k and or different mixture models. The user must choose a model among the possible models, no formal method to do this is given. In later work, Raftery and Fraley [25], the AWE statistic is replaced by an alternative approximation called the Bayesian information criterion (BIC) of Schwarz [57]:

$$-2 \log L(\hat{\Psi}) + p \log n \tag{2.3}$$

where $\Psi = (w_1, w_2, \ldots w_{k-1}, \boldsymbol{\theta})$ and p is the dimension of $\boldsymbol{\theta}$. Dasgupta and Raftery [18]

obtained good results in a examples with constant-shape Gaussian models and the BIC to determine the number of clusters. Fraley and Raftery [25], explain their approach can form the basis of a more general model-based strategy for clustering. The method needs to specify a maximum number of clusters to consider, M, and a set of possible parametrisations of the covariance matrix to consider. Then an agglomerative clustering is carried out for the unconstrained Gaussian model and an initial classification into M clusters is obtained and used as initial classification to run the EM algorithm. Finally, the BIC is used to look for the combination of number of clusters and model that best fits the data.

There are some aspects about the EM algorithm that need to be considered. The rate of convergence near the optimum can be very slow and it may not be practical for models with very large number of components. The algorithm would have numerical problems if at least one of the covariance matrices is close to singularity.

Fraley and Raftery [26], [25] considered a constant Poisson process to model noisy data. The mixture likelihood is then

$$L(\Psi|\mathbf{y}) = \prod_{i=1}^{n} \left[\frac{w_0}{V} + \sum_{j=1}^{k} w_j N_p(y_i|\boldsymbol{\theta}_j) \right], \tag{2.4}$$

where V is the hypervolume of the data region and $\sum_{j=0}^{k} w_j = 1$ and $\boldsymbol{\theta}_j = (\boldsymbol{\mu}_j, \Sigma_j)$. An observation contributes $1/V$ if it is part of the noise, otherwise it contributes a Gaussian term.

The basic procedure initially obtains an estimate for the noise, then hierarchical clustering is applied to denoised data to obtain an initial partition. The EM used for the augmented model (2.4) is initialised with the hierarchical clustering partition and the result of the denoising procedure for the noise component. The EM works well with a good initial identification of the noise and clusters.

Bensmail *et al* [4] presented a fully Bayesian approach whereby inference was made through Gibbs sampling. They used conjugate prior distributions for Ψ, the parameters of the mixture model. The prior distribution of the mixing proportions is a Dirichlet distribution $\mathbf{w} \sim D(\alpha_1, \ldots, \alpha_k)$. The prior distributions for the means, conditionally on the covariance matrices, Σ_k, are Gaussian, $\boldsymbol{\mu}_k|\Sigma_k \sim N_p(\xi_k, 1/\tau_k\Sigma_k)$. The conjugate prior of the covariance matrix depends on the model that is being considered and it assumes one of the following forms: λI, $\lambda_k I$, Σ, $\lambda_k \Sigma$, $\lambda D_k A D_k'$, $\lambda_k D_k A D_k'$, $\lambda_k D A_k D'$

or Σ_k.

In this case, the model and the number of components is selected through an approximation of the Bayes factors. Consider a model as the combination of a number of components and a form of the covariance matrix, the Bayes factor for M_i against model M_j given the data $\mathbf{y}$ corresponds to the ratio of posterior to prior odds given by

$$B_{ij} = p(\mathbf{y}|M_i)/p(\mathbf{y}|M_j), \tag{2.5}$$

where

$$p(\mathbf{y}|M_l) = \int p(\mathbf{y}|\boldsymbol{\theta}_l, M_l)p(\boldsymbol{\theta}_l|M_l)d\boldsymbol{\theta}_l, \tag{2.6}$$

where $\boldsymbol{\theta}_l$ is the vector of parameters for model M_l, $p(\boldsymbol{\theta}_l|M_l)$ is the prior density $(l = i, j)$, equation (2.6) is called the *integrated likelihood.* A good review on Bayes factors can be found in Kass and Raftery [39]. The main computational challenge here is to approximate the integrated likelihood. The Gibbs sampler output is used to obtain the Laplace-Metropolis estimator of the integrated likelihood, Raftery [50]. The Laplace method for the integral of a real-valued function $f(u)$, where u is a p-dimensional vector, yields the approximation

$$p(D) \approx (2\pi)^{\frac{p}{2}}|H(\Psi)|^{\frac{1}{2}}p(\mathbf{y}|\tilde{\boldsymbol{\theta}})p(\tilde{\boldsymbol{\theta}}) \tag{2.7}$$

where p is the dimension of $\boldsymbol{\theta}$, $\tilde{\boldsymbol{\theta}}$ is the posterior mode of $\boldsymbol{\theta}$, and $H(\Psi)$ is minus the inverse Hessian of $h(\boldsymbol{\theta}) = \log\{p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})\}$, evaluated at $\boldsymbol{\theta} = \tilde{\boldsymbol{\theta}}$.

The derivatives this approximation requires are not easily available, therefore the posterior simulations are used to obtain robust estimates of the quantities needed to compute expression (2.7). Finally, the Bayes factors are obtained as the ratios of the integrated likelihoods as given in equation (2.5). The proposed Bayesian cluster analysis worked well for several examples but only the first four forms of the covariance matrices given above were considered.

2.3 Bayesian semi-parametric approach to model-based clustering

Another approach proposed in literature for density estimation is Bayesian inference using mixtures of Dirichlet processes. Escobar and West [23] describe the normal mixture model as follows. They assume that data $Y_1, Y_2, \ldots, Y_n$ are conditionally independent and normally distributed, $(Y_i|\theta_i) \sim N(\mu_i, V_i)$, $\theta_i = (\mu_i, V_i)$ for $i = 1, \ldots, n$. The parameters θ_i come from some prior distribution $G(\cdot)$ on $\mathbb{R} \times \mathbb{R}^+$. If $G(\cdot)$ is uncertain and modelled as a Dirichlet process, then the data can be seen as coming from a Dirichlet mixture of normals (Escobar [22], Ferguson [24], Antoniak [2])

$$\begin{aligned} \theta_i &\sim G, \\ G &\sim Dir(\alpha G_0(\cdot)). \end{aligned}$$

In particular they consider a Dirichlet process defined by a positive scalar α and $G_0(\cdot)$, the prior guess at the shape, a specified bivariate distribution over $\mathbb{R} \times \mathbb{R}^+$. $G_0(\cdot)$ is the prior expectation of $G(\cdot)$. An important part of the model structure is associated with the discreteness of $G(\cdot)$ under the Dirichlet process assumption. In any sample θ of size n from $G(\cdot)$ there is a positive probability of coincident values. For any $i = 1, \ldots, n$, consider $\theta^{(i)} = \{\theta_1, \ldots, \theta_{i-1}, \theta_{i+1}, \ldots, \theta_n\}$. The conditional prior for $(\theta_i|\theta^{(i)})$ is

$$(\theta_i|\theta^{(i)}) \sim \alpha a_{n-1} G_0(\theta_i) + a_{n-1} \sum_{j=1, j\neq i}^{n} \delta_{\theta_j}(\theta_i) \tag{2.8}$$

where $\delta_{\theta_j}(\theta_i)$ denotes a unit point mass at $\theta_j = \theta_i$ and $a_r = 1/(\alpha + r)$ for positive integers r. The distribution of $(\theta_{n+1}|\theta)$ is

$$(\theta_{n+1}|\theta) \sim \alpha a_n G_0(\theta_{n+1}) + a_n \sum_{i=1}^{n} \delta_{\theta_i}(\theta_{n+1}). \tag{2.9}$$

Therefore, given θ, the next case θ_{n+1} represents a new distinct value with probability αa_n and is otherwise drawn uniformly among the first n values. This is the case for the first n values as well, so with positive probability they will reduce to some $k < n$ distinct values. The prior mean $G_0(\cdot)$ needs to be specified, a normal/inverse-gamma form is convenient for this model.

The corresponding prior distribution for k, see Antoniak [2], depends critically on α. It also depends on n and it is unimodal. One can select the value of α that, given n, will put the modal value where desired. To deal with this issue Lo [41] proposed to randomize the value of α. Escobar and West [23] show how to sample from the posterior of α at each stage of the Gibbs sampler simulation.

West *et al* [66] use this scheme to analyse multivariate data, particularly the five dimensional version of the Lubischew's [42] beetle data set. In his results the $k = 3$ and $k = 4$ are given the highest posterior probability. The number of species in the data set is actually 3. Although the paper concentrates on density estimation, the authors suggest this could be used as a clustering strategy. However, it is difficult to evaluate the effect that the parameter α has on posterior inference, particularly on the posterior distribution of k, the number of groups. As Green and Richardson [35] point out, a difficulty in the DP model is that a single parameter controls variability, making it difficult for the user to specify it *a priori*. This approach would require a detailed analysis on this aspect.

Quintana and Iglesias [49] give a decision theoretic formulation of product partition models (PPM's). The models associated to observations $\mathbf{y} = \mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n$ are represented by $\mathcal{M} = \{M_\rho : \rho \in \mathcal{P}\}$, where $\mathcal{P}$ is the set of all partitions of $S_0 = \{1, \ldots, n\}$. The underlying partition structure in this model is related to the structure that arises in connection with Dirichlet processes. The authors present a procedure to select a partition model with a specified purpose. The algorithm is based on the particular decision problem at hand with the loss function chosen to reflect the corresponding aspects of interest, for example, estimating some parameter that determines the distribution of $\mathbf{y}$, detecting outliers, exploring some specific posterior distribution of interest.

Minimisation of the given loss function needs a partition ρ^* that attains the corresponding optimal value to be found. A clustering algorithm is introduced to find such a partition. At the beginning of a given step the current partition is $\rho_j = \{S_1^j, \ldots, S_{|\rho_j|}^j\}$, where S_1^j contains all those units that have not yet been selected. The algorithm finds the most outlying observation $k^* \in S_1^j$. Then all the possible partitions generated by separating k^* from S_1^j are evaluated, including the possibility that k^* defines a new cluster with only that observation. If none of the partitions $|\rho_j|$ are better than ρ_j the algorithm stops proposing the partition ρ_j, otherwise it is replaced by the better partition and the whole process is repeated.

2.4 Other approaches

Projection pursuit is a technique that has been used in cluster analysis. Low dimensional projections of the multivariate data are used to obtain interesting views of the full-dimensional data. Clusters are found by minimising a projection index, for example, if the projection produces non-normal distributions, any test for non-normality could be used as a projection index.

Bolton and Krzanowski [7] proposed a clustering method based on projection pursuit and non-hierarchical clustering methods. They introduce a projection pursuit index that takes into account the scale in the data. They propose diagnostics for finding the number of groups in the projected data based on within group sums-of-squares.

Peña and Prieto [47] propose a one-dimensional projection pursuit algorithm based on directions obtained by both maximising and minimising the kurtosis coefficient of the projected data. Once a first direction is chosen, the data are projected in this direction. They evaluate if the projected points can be split into clusters along this first direction. Assuming that the set $S = (\mathbf{y}_1, \ldots, \mathbf{y}_n)$ is split into k non overlapping sets $S = S_1 \cup S_2 \cup \cdots \cup S_k$, where $S_i \cap S_j = \emptyset$ for all i, j, $i \neq j$, the sample data are projected over a second direction checking if any cluster S_i, $i = 1, \ldots, k$ can be further split. The procedure is repeated until the data are split into m sets, in the projections in one direction some clusters might mask the presence of other clusters.

Peña and Prieto [47] consider as interesting directions those in which the projected data cluster around different means. These means are well separated with respect to the mean variability of the distribution of the points around their means. The criteria used to identify the clusters is based on the analysis of the sample spacings or first-order gaps between ordered statistics of the projections. If the data come from a univariate distribution, then large gaps should appear near the extremes of the distributions and small gaps near the center. This pattern would be altered by the presence of clusters. They consider a set of observations can be split into two clusters when they find a sufficiently large first-order gap in the sample. After completing the analysis of the gaps, the algorithm assigns observations within the clusters identified in the data. The main aim of the final step is to check if the observations suspected to be outliers are just a product of the choice of directions. This procedure is based on standard multivariate tests using the Mahalanobis distance. They have found this algorithm effective in

practice.

Posse [48] uses a hierarchical agglomerative clustering, starting not from the usual singleton, but from an efficient classification of the data in many groups. The initial classification is derived from a subgraph of the minimum spanning tree (MST) (see section 4.3 for a detailed description of the MST) associated to the data. The subgraph is obtained by first trimming out its longest edges, which separates isolated observations in the periphery of the clusters and disconnects any well separated clusters in the sample. To determine the number of edges to consider in this peeling, he compares the empirical distributions of the longest edges with their theoretical distributions when data come from an homogeneous Gaussian population. The remaining connected components in the MST are then separated into smaller groups of roughly the same size. The average size of these new components is determined by the number of edges.

2.5 Deciding on the number of clusters

A major difficulty in cluster analysis is the choice of the number of clusters to consider. Tibshirani *et al* [62] propose a method for estimating the number of clusters based on the " gap statistic ". Let $\{y_{ij}\}$ be the data, with $i = 1, \ldots, n$ observations and $j = 1, \ldots, p$ features measured. Let $d_{ii'}$ be the Euclidian distance between observations i and i'. When the data have been clustered into k groups, let $C_1, \ldots, C_k$ denote the indices of observations in C_r, $r = 1, \ldots, k$ and $n_r = |C_r|$. Let

$$D_r = \sum_{i,i' \in C_r} d_{ii'}$$

be the sum of the pairwise distances of all points in cluster r and

$$W_k = \sum_{r=1}^{k} \frac{1}{2n_r} D_r,$$

where W_k is the pooled within cluster sum of squares around the cluster means if distance d is the squared Eucledian distance.

They propose to standarise the graph of $log(W_k)$ by comparing it with its expectation under an appropriate null reference distribution of the data. The estimate for the

optimal number of clusters k they use is

$$Gap_n(k) = E_n^* log(W_k) - log(W_k),$$

where E_n^* denotes the expectation under a sample of size n from the reference distribution. To construct the reference distribution, they model the components as log-concave densities. They either generate each reference feature uniformly over the range of observed values for that feature or from a uniform distribution over a box aligned with the principal components of the data. Taking into account the variation over the simulation of the reference distribution they propose as the optimal value of the number of clusters the smallest k for which

$$Gap_n(k) \geq Gap_n(k+1) - s_{k+1},$$

where s_{k+1} denotes the standard deviation of the simulated value from the reference distribution. They conclude the gap statistic is efficient for well-separated clusters.

Sahu and Russell [56] consider the problem of determining the unknown number of components in mixture models. They propose the use of a distance measure to compare two distributions. They focus on the Kullback-Leibler (KL) measure because of the advantages it has in terms of simplicity and analytical tractability. The method starts with a large value of k and reduces the number of components sequentially by collapsing until no further collapsing can be done without considerable loss in the fit. That is, they consider the merging of the two components rather than carrying out a re-fit of the model. Once they have computed the distance between the original model and all the C_2^k collapsed versions, they select the best version of the model with $k-1$ components. The parameters for the new components are calculated keeping fixed the parameters that correspond to the other components. Then the posterior probability

$$P_{ck}(k) = Pr\{d(f^{(k)}, f^{(k-1)}) \leq c_k | data\}$$

is evaluated and the collapsed version is kept if $P_{ck} > \alpha$. In that case, they replace $f^{(k)}$ by $f^{(k-1)}$. A re-fit of the whole model is carried out at the end of the process to ensure that the final posterior distribution has been adequately identified.

They use accurate approximations of the KL distance based on numerical quadra-

tures between the model with k components and its collapsed version and they also propose an alternative weighted KL distance with similar properties to the KL distance. The authors suggest this collapsed scheme could also be used in the RJMCMC sampler.

2.6 Clustering for gene data

Recently the study of genomes, their sequence and function, has become an active area of research in bioinformatics. DNA microarrays [1] and oligonucleotide arrays generate large amounts of data in different types of gene studies. As a result there is an eminent need to develop analytical methodologies to extract useful information from these large data sets. The process to obtain these data is not simple. Problems arise at different stages and bias could be introduced as a result of the microarray technology.

Gene expression data is being used for different purposes such as examining changes in gene expression at different stages in the cell cycle or during embryonic development, assigning probable biological functions to newly discovered genes by comparison with the expression patterns of known genes, identifying new targets for therapeutic drugs and in disease diagnosis with a view to individualized prognosis and therapy.

In particular, the problem concerned with the assignment of a biological function in cellular signalling has made clustering a useful tool in the exploratory analysis of microarray data. There is a growing number of clustering algorithms that are being proposed to analyse gene expression data. These algorithms include hierarchical clustering , self-organising maps, k-means, graph-theoretic approaches, among others. Many of these have been reported to give successful insight to gene expression data. However, as pointed out by some authors, none of these strategies provide a measure of uncertainty on the classification on quantify cluster membership probabilistically. Model-based clustering has been proposed to analyse gene expression data by several authors, see for example Yeung *et al* [68], McLachlan *et al* [44] and Wakefield *et al*

[1] Microarrays exploit the preferential binding of complementary single-stranded nucleic-acid sequences. The underlying principle is the same for all microarrays, regardless of how they are made, the unknown sample is hybridized to an ordered array of immobilized DNA whose sequence is known. RNA from two different tissues or cell populations is used to synthesize single-stranded cDNA in the presence of nucleotides labelled with two different dyes (for example, Cy3 and Cy5). Both samples are mixed in a small volume of hybridization buffer and hybridized to the array surface, usually by stationary hybridization under a cover-slip, resulting in competitive binding of differentially labelled cDNAs to the corresponding array elements. High-resolution confocal fluorescence scanning of the array with two different wavelengths corresponding to the dyes used provides relative signal intensities and rations of mRNA abundance for genes represented on the array.

[65].

CHAPTER 3

Cluster analysis using RJMCMC

We have discussed in Chapter 2 the use of Gaussian finite mixture models in model-based clustering. We assume that the observed data belong to one of K clusters which comprise the target population in a certain proportion. The number of groups is associated one-to-one with the k components of the mixture model. We assume there is no prior certainty about k, the number of components and therefore it is considered as an additional parameter.

A Bayesian formulation of the problem allows inference to be based on the joint posterior distribution of the number of components in the mixture and the other unknown parameters given the data. The Reversible Jump Markov chain Monte Carlo (RJMCMC) procedure proposed by Green [32] has been central in extending the use of Markov chain Monte Carlo (MCMC) to problems where the parameter space is of variable dimension. Richardson and Green [51] have analysed the use of RJMCMC to fit a univariate mixture of normal distributions which essentially addresses one-dimensional model-based cluster analysis.

In this chapter we will describe an extension of the latter to higher dimensions. We will not consider any restriction on the parameters of the normal components, particularly for the covariance matrices Σ_k. The performance of the sampler will be evaluated through some examples and we will discuss the posterior inference for the number of components as well as the posterior classification. The algorithm we discuss in this chapter will often produce empty components which may or may not be close to the observed data. For the cluster analysis, in contrast to what one would do in a density estimation context, we will dispense with the empty components when it is required for the analysis. This situation will occur again in the following chapters and

once again the empty components will be removed when it is required for the analysis. We will present a sensitivity analysis of the posterior inference to the prior assumptions.

3.1 The one dimensional problem

With the setting given in section 1.2.1, following Richardson and Green [51] we will describe the mixture model as expressed in equation (1.3) where there is only one feature observed for the sampled data and all densities belong to the same parametric family. The unknown parameters $\Psi = (k, \mathbf{w}, \boldsymbol{\theta})$ are assumed to be drawn from appropriate prior distributions. Then, the joint distribution of all variables can be written as

$$p(k, \mathbf{w}, \mathbf{z}, \boldsymbol{\theta}, \mathbf{y}) = p(k)p(\mathbf{w}|k)p(\mathbf{z}|\mathbf{w}, k)p(\boldsymbol{\theta}|\mathbf{z}, \mathbf{w}, k)p(\mathbf{y}|\boldsymbol{\theta}, \mathbf{z}, \mathbf{w}, k), \tag{3.1}$$

where $p(\cdot|\cdot)$ denotes generic conditional distributions.

Imposing further conditional independence, so that $p(\boldsymbol{\theta}|\mathbf{z}, \mathbf{w}, k) = p(\boldsymbol{\theta}|k)$ and $p(\mathbf{y}|\boldsymbol{\theta}, \mathbf{z}, \mathbf{w}, k) = p(\mathbf{y}|\boldsymbol{\theta}, \mathbf{z})$, then equation (1.4) simplifies to

$$p(k, \mathbf{w}, \mathbf{z}, \boldsymbol{\theta}, \mathbf{y}) = p(k)p(\mathbf{w}|k)p(\mathbf{z}|\mathbf{w}, k)p(\boldsymbol{\theta}|k)p(\mathbf{y}|\boldsymbol{\theta}, \mathbf{z}), \tag{3.2}$$

where we have $p(\mathbf{y}|\boldsymbol{\theta}, \mathbf{z}) = f(\cdot|\boldsymbol{\theta}_{z_i})$ and $p(z_i = j) = w_j$, for $j = 1, \ldots, n$. Another layer to the hierarchy was proposed by the authors, allowing the priors for k, w and $\boldsymbol{\theta}$ to depend on hyperparameters λ, δ and γ so that these are drawn from independent hyperpriors. Then the joint distribution of all variables is given by

$$p(\lambda, \delta, \gamma, k, \mathbf{w}, \mathbf{z}, \boldsymbol{\theta}, \mathbf{y}) = p(\lambda)p(\delta)p(\gamma)p(k|\lambda)p(\mathbf{w}|k, \delta)p(\mathbf{z}|\mathbf{w}, k)p(\boldsymbol{\theta}|k, \gamma)p(\mathbf{y}|\boldsymbol{\theta}, \mathbf{z}). \tag{3.3}$$

3.2 The normal mixture model

For the specific case where all the components in the mixture model are normally distributed, for any $f(y_i|\boldsymbol{\theta}_j)$, $\boldsymbol{\theta}_j = (\mu_j, \sigma_j^2)$, $j = 1, \ldots, k$. Richardson and Green [51] adopted an independent sample structure for the parameters μ_j and σ_j^2 and consider a " weakly informative " prior structure

$$\mu_j \sim N(\xi, \kappa^{-1}), \tag{3.4}$$

$$\sigma_j^{-2} \sim \Gamma(\alpha, \beta), \tag{3.5}$$

where the parametrisation of the gamma distribution has mean α/β and variance α/β^2. The normal prior distribution for μ_j is defined over an interval of variation, this interval could be proposed *a priori* or the observed range could be used. Thus, ξ is taken as the midpoint of the observed range and κ is a multiple of $1/R^2$ where R is the length of the observed interval of variation.

To avoid restricting the size of the σ_j^2 by the extent of the range of the data, another level of hierarchy was introduced by allowing β to be drawn from a gamma distribution with parameters (g, h), where $\alpha > 1 > g$. In particular, values $\xi = 1/R^2$, $\alpha = 2$, $g = 0.2$ and $h = 10/R^2$ were used.

To deal with the identifiability problem introduced in section 1.2.3, the component means μ_j are assumed to be in increasing numerical order, $\mu_1 < \mu_2 < \ldots < \mu_k$. The prior on the mixing proportions $\mathbf{w}$ is always taken as a symmetric Dirichlet distribution,

$$\mathbf{w} \sim D(\delta, \delta, \ldots, \delta) \tag{3.6}$$

in this case δ is held fixed with $\delta = 1$.

The prior distribution for the number of components, for practical convenience, was taken to be uniform between 1 and a maximum prespecified number of components k_{max}. Richardson and Green [51] emphasised that results could be converted to other priors on these values by means of the identity

$$p^*(k, \boldsymbol{\theta}_k|y) \propto p(k, \boldsymbol{\theta}_k|y)\frac{p^*(k)}{p(k)}.$$

3.3 Reversible jump methods

Section 1.7.1 gave the general form of the reversible jump MCMC algorithm. It establishes the need to define a family of moves that allow the sampler to move between different dimensions. When the sampler attempts a move to a different dimension, a set of random numbers and a deterministic function must be defined to obtain the corresponding values of the required parameters.

The moves used to define the reversible jump sampler for the hierarchical model described in the previous section consisted of a split/merge move and a birth/death move. A sweep to update the parameters in the current RJMCMC iteration involved the following:

(i) Updating the weights $\mathbf{w}$, which can be done using the full posterior conditional in a Gibbs step. Through conjugacy, the posterior conditional distribution of the weights remains a Dirichlet distribution given by

$$\mathbf{w}|\ldots \sim D(\delta + n_1, \delta + n_2, \ldots, \delta + n_k), \tag{3.7}$$

where $n_j = \#\{i : z_i = j\}$, that is the number of observations currently allocated to the $j-$th component . Here "$|\ldots$" denotes conditioning on all other variables.

(ii) Updating parameters $\{\mu_j\}$ and $\{\sigma_j\}$, which can also be done in a Gibbs step. The conjugated posterior full conditional distributions are

$$\mu_j|\ldots \sim N\left\{\frac{\sigma_j^{-2}\sum_{i:z_i=j} y_i + \kappa\xi}{(\sigma_j^{-2}n_j + \kappa)}, (\sigma_j^{-2}n_j + \kappa)^{-1}\right\} \tag{3.8}$$

$$\sigma_j^{-2}|\ldots \sim \Gamma\left\{\alpha + \frac{1}{2}n_j, \beta + \frac{1}{2}\sum_{i:z_i=j}(y_i - \mu_j)^2\right\} \tag{3.9}$$

(iii) Updating the allocation variables $\mathbf{z}$. Here the probability of the i-th observation to be allocated in the j-th component is proportional to

$$pr(z_i = j) \propto \frac{w_j}{\sigma_j}\exp\left\{-\frac{(y_i - \mu_j)^2}{2\sigma_j^2}\right\}. \tag{3.10}$$

for $i = 1, \ldots, n$.

(iv) Updating the hyperparameter β. This is also done in a Gibbs step using the resulting full conditional gamma distribution

$$\beta|\ldots \sim \Gamma\left(2g + 2k\alpha, h + \sum_j \sigma_j^{-2}\right) \tag{3.11}$$

(v) Attempting to split one of the components in the current iteration into two or to combine two components into one.

For this move, the probabilities of attempting to split or combine components are b_k and $d_k = 1 - b_k$ respectively, depending on k. As one might expect, these probabilities are taken as $b_{k_{max}} = d_1 = 0$, $d_{k_{max}} = b_1 = 1$ and $b_k = d_k = 0.5$ for all other values of k.

For the combine proposal, two adjacent components, denoted (j_1, j_2), are selected at random. Here, the adjacency refers to adjacency in terms of the current values of the component means. Once they are selected, these two adjacent components are merged into a new one denoted j_*. In this way the identifiability constraint which requires $\mu_1 < \mu_2 < \ldots < \mu_k$, is satisfied. Parameter values for the new component need to be computed. In order to do this Richardson and Green [51] proposed to equate the first three moments,

$$\begin{aligned} w_{j_*} &= w_{j_1} + w_{j_2}, \\ w_{j_*}\mu_{j_*} &= w_{j_1}\mu_{j_1} + w_{j_2}\mu_{j_2}, \\ w_{j_*}(\mu_{j_*}^2 + \sigma_{j_*}^2) &= w_{j_1}(\mu_{j_1}^2 + \sigma_{j_1}^2) + w_{j_2}(\mu_{j_2}^2 + \sigma_{j_2}^2). \end{aligned}$$

For the reverse split move, three random numbers are generated using beta distributions $u_1 \sim be(2,2)$, $u_2 \sim be(2,2)$, $u_3 \sim be(1,1)$. The six required parameters that satisfy the above moment matching are

$$\begin{aligned} w_{j_1} &= u_1 w_{j_*}, \\ w_{j_2} &= (1-u_1)w_{j_*}, \\ \mu_{j_1} &= \mu_{j_*} - u_2\sigma_{j_*}\sqrt{\frac{w_{j_1}}{w_{j_2}}}, \\ \mu_{j_2} &= \mu_{j_*} + u_2\sigma_{j_*}\sqrt{\frac{w_{j_2}}{w_{j_1}}}, \\ \sigma_{j_1}^2 &= u_3(1-u_2^2)\sigma_{j_*}^2\frac{w_{j_*}}{w_{j_1}}, \\ \sigma_{j_2}^2 &= (1-u_3)(1-u_2^2)\sigma_{j_*}^2\frac{w_{j_*}}{w_{j_2}}, \end{aligned}$$

provided $\mu_{j_1} < \mu_{j_2}$.

(vi) The birth of a new empty component or the death of an existing empty component. A random choice to attempt a birth or a death is made using the same probabilities b_k and d_k as for the previous move.

For a birth, the parameters of the new component are drawn using the prior

specifications

$$\begin{aligned} w_{j_*} &\sim be(1, g), \\ \mu_{j_*} &\sim N(\xi, \kappa^{-1}), \\ \sigma_{j_*}^{-2} &\sim \Gamma(\alpha, \beta). \end{aligned}$$

The weights of the existing components must be rescaled so that condition given by expression (1.2) is satisfied.

For the death move, a random choice from the existing empty components is made and the weights of the remaining components are rescaled to satisfy expression (1.2).

The acceptance probabilities for the split/combine and birth/death moves are computed from equation (1.20), see Richardson and Green [51], so detailed balance is satisfied. The authors point out that the chain is irreducible since it can move from any value of k to any other value in steps of one at a time, allocations of observations have all positive probability and the parameters and hyperparameters are sampled from distributions whose supports are the natural parameter spaces. Also the chain is aperiodic since given any arbitrarily small neighbourhood of a current state, after a sweep of the sampler there is a positive probability that the sampler lies in that neighbourhood.

In general, for the one dimensional problem, the reversible jump proved to be a powerful technique to generate a sample from the joint posterior distribution of all unknown variables. The complexity of the process in applications requires considerable tuning. Some aspects related to sensitivity of the posterior inference to prior assumptions were highlighted by the authors and they will be taken into account as we proceed to extend their methodology to higher dimensional problems.

3.4 Multivariate extension

In the present section, we describe an extension of the RJMCMC method of Richardson and Green to classify multivariate observations into an unknown number of clusters. The univariate implementation, as presented in the previous section, is followed, pointing out the significant variations.

Again consider a mixture model of normal multivariate distributions. Let the

p-vectors $\mathbf{y}_1, \ldots \mathbf{y}_n$ be the observed values of the corresponding random variables $\mathbf{Y}_j$. The density function for $\mathbf{Y}_j$ is then assumed to be a k-component normal mixture of the form

$$f(\mathbf{y}_j) = \sum_{i=1}^{k} w_i N_p(\boldsymbol{\mu}_i, \Sigma_i). \tag{3.12}$$

We also adopt an independent sample structure for the parameters $\boldsymbol{\mu}_i$ and Σ_i. As proposed by Stephens [59], following Richardson and Green [51], a natural generalisation for the p-dimensional case is obtained replacing univariate normal prior distributions with multivariate normal distributions and gamma prior distributions with Wishart distributions. Hence the prior distributions for the parameters $\mathbf{w}$, $\boldsymbol{\mu}_i$ and Σ_i are respectively Dirichlet, multivariate normal and inverse Wishart, specifically

$$\begin{aligned} \mathbf{w} &\sim D(\delta, \delta, \ldots, \delta) \\ \boldsymbol{\mu}_i &\sim N_p(\xi, \kappa^{-1}) \\ \Sigma_i^{-1} &\sim W_p(2\alpha, (2\beta)^{-1})) \end{aligned} \tag{3.13}$$

for $i = 1, 2, \ldots, k$. Where κ, β and h are $p \times p$ matrices, ξ is a $p \times 1$ vector and α, δ and g are scalars.

The prior distribution for the number of components, k, is taken to be uniform in $\{1, \ldots, k_{max}\}$, where $k_{max} = 30$. As before, to reflect the belief that the variances of the components in the mixture are similar but without restricting them to be equal, another level in the hierarchical model is given. The hyperprior distribution for the scale matrix β is a Wishart distribution

$$\beta \sim W_p(2g, (2h)^{-1}). \tag{3.14}$$

The notation $W_p(\nu, A)$ is used for a Wishart distribution in p dimensions with parameters ν and A. Usually, a Wishart distribution is given for $\nu \geq p$, where ν is an integer. This is useful when ν is interpreted as a sample size and the sample covariance has a Wishart distribution. However, a density can be defined by a non-integer ν provided $\nu > p - 1$. This Wishart distribution has density

$$W_p(V; \nu, A) = K|A|^{-\nu/2}|V|^{(\nu-p-1)/2} exp\left\{-\frac{1}{2}\mathrm{tr}(A^{-1}V)\right\} I(V \text{positive definite}), \tag{3.15}$$

on the space of all symmetric matrices, where $I(\cdot)$ denotes an indicator function and

$$K^{-1} = 2^{(\nu p)/2}\pi^{(p(p-1))/4}\prod_{c=1}^{p}\Gamma\left(\frac{\nu+1-c}{2}\right).$$

The values $\alpha = 2$ and $g = 0.2$ were proposed by Richardson and Green [51] to define the corresponding gamma hyperprior for β in the one dimensional case. Stephens [59] suggested that a slightly stronger constraint was needed to define the hyperprior for β, and he successfully used $\alpha = 3$ and $g = 0.3$ for the bivariate case. This gives an improper hyperprior on β. Here $W_p(\nu, A)$ denotes a density proportional to expression (3.15) for $\nu \leq p-1$. However, the posterior distribution is seen to be proper for the two dimensional case. When the dimension of the observed variables increases, in order to have a proper posterior, the values $\alpha = p+1$ and $g = \alpha/10$ will be considered. Sensitivity of the posterior inference to this assumption will be assessed.

Finally, the values given to the remaining constants are

$$\begin{aligned}
\xi &= (\xi_1, \ldots, \xi_p) \\
\kappa &= \begin{pmatrix} \frac{1}{R_1^2} & 0 & \ldots & 0 \\ 0 & \frac{1}{R_2^2} & \ldots & 0 \\ 0 & & \ddots & 0 \\ 0 & \ldots & 0 & \frac{1}{R_p^2} \end{pmatrix} \\
h &= \begin{pmatrix} \frac{100g}{\alpha R_1^2} & 0 & \ldots & 0 \\ 0 & \frac{100g}{\alpha R_2^2} & \ldots & 0 \\ 0 & & \ddots & 0 \\ 0 & \ldots & 0 & \frac{100g}{\alpha R_p^2} \end{pmatrix} \\
\delta &= 1
\end{aligned}$$

where ξ_i is the midpoint of the observed intervals of variation for the data in the ith variable and R_i is the respective length of that interval, for $i = 1, \ldots, p$.

A reversible jump sampler will be used to sample values from the target distribution, the sequence of steps within each MCMC iteration is as follows:

(i) Updating the weights $\mathbf{w}$, which can be done, as in the one-dimensional case, in a

Gibbs step from the posterior full conditional

$$\mathbf{w}|\ldots \sim D(\delta + n_1, \delta + n_2, \ldots, \delta + n_k), \tag{3.16}$$

where $n_i = \#\{i : z_j = i\}$.

(ii) Updating parameters $\boldsymbol{\mu}_i$ and Σ_i, which can also be done in a Gibbs step using the resulting conjugate full conditional distributions

$$\boldsymbol{\mu}_i|\ldots \sim N_p\big((n_i\Sigma_i^{-1} + \kappa)^{-1}(n_i\Sigma_i^{-1}\overline{y}_i + \kappa\xi), (n_i\Sigma_i^{-1} + \kappa)^{-1}\big) \tag{3.17}$$

$$\Sigma_i^{-1}|\ldots \sim W_p\left\{2\alpha + n_i, [2\beta + \sum_{i:z_j=i}(y_j - \boldsymbol{\mu}_i)(y_j - \boldsymbol{\mu}_i)^T]^{-1}\right\} \tag{3.18}$$

(iii) Updating the allocation variables z, where

$$pr(z_j = i) \propto w_i N_p(\mathbf{y}_j; \boldsymbol{\mu}_i, \Sigma_i). \tag{3.19}$$

(iv) Updating the hyperparameter β, also done using the posterior full conditional Wishart distribution,

$$\beta|\ldots \sim W_p\left(2g + 2\alpha k, [2h + 2\sum_i \Sigma_i^{-1}]^{-1}\right) \tag{3.20}$$

(v) Attempting to split one of the components into two or to combine two into one. The probability of attempting to split or combine components is exactly the same as for the one-dimensional case.

For the combine proposal, the univariate case considered choosing at random a pair of components (j_1, j_2) that were adjacent in terms of the means. That is, $\mu_{j_1} < \mu_{j_2}$ with no other μ_j in the interval $[\mu_{j_1}, \mu_{j_2}]$. We have not considered any partial ordering on the means for the higher dimensional case, two components (j_1, j_2) are selected at random with probability proportional to the inverse of the Mahalanobis distance between every pair of means.

The Mahalanobis distance between $\boldsymbol{\mu}_i$ and $\boldsymbol{\mu}_j$ is expressed as

$$d(i,j) = (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^T \hat{\Sigma}^{-1} (\boldsymbol{\mu}_i - \boldsymbol{\mu}_j) \quad \text{where} \tag{3.21}$$

$$\hat{\Sigma} = \frac{w_i \Sigma_i + w_j \Sigma_j}{w_i + w_j}. \tag{3.22}$$

Once the two components are selected, they are merged into a new component labelled j_*.

Unfortunately, the direct extension of the univariate deterministic function to compute the covariance matrix of the new components was not convenient in terms of the efficiency of the sampler. If the first three moments were considered, namely

$$\begin{aligned} w_* &= w_1 + w_2, \\ w_* \boldsymbol{\mu}_{i_*} &= w_1 \boldsymbol{\mu}_{i_1} + w_2 \boldsymbol{\mu}_{i_2}, \\ w_* (\boldsymbol{\mu}_* \boldsymbol{\mu}_*^T + \Sigma_*) &= w_1 (\boldsymbol{\mu}_1 \boldsymbol{\mu}_1^T + \Sigma_1) + w_2 (\boldsymbol{\mu}_2 \boldsymbol{\mu}_2^T + \Sigma_2), \end{aligned}$$

then, in almost every case, this approach produced a matrix which was not positive definite. Some other conditions on the elements of the covariance matrices of the selected components were explored to allow a larger number of positive definite matrices to be obtained while attempting the split/combine move. For example, the 2^{nd} moment was replaced by the equality $w_* \Sigma_* = w_1 \Sigma_1 + w_2 \Sigma_2$, in which case a diagonal matrix of random numbers has to be generated. However, for the new matrix to remain a positive definite matrix, the random numbers needed to be very small, which made the coverage of the parameter space inadequate. Other conditions on the eigenvalue decomposition of the covariance matrices were tried, but the reversibility of the move was not easily achieved. Recently Dellaportas and Papageorgiou [19] presented a split/combine move using the spectral decomposition of the covariance matrix. In their work they show that the RJMCMC sampler works but do not give any details on the performance of the sampler. Zhihua *et al* [69] preserved the first two moments by considering that covariance matrices for all components share a common eigenvector matrix.

To carry out the cluster analysis using a RJMCMC, the functions used to merge the two selected components into a new j_*, which we found more useful, are

$$\begin{aligned} w_* &= w_1 + w_2, \\ w_*\boldsymbol{\mu}_{i_*} &= w_1\boldsymbol{\mu}_{i_1} + w_2\boldsymbol{\mu}_{i_2}, \\ w_*\sigma^2_{ii_*} &= w_1\sigma^2_{ii_1} + w_2\sigma^2_{ii_2}, \\ w_*\frac{\sigma_{ij_*}}{\sigma_{ii_*}\sigma_{jj_*}} &= w_1\frac{\sigma_{ij_1}}{\sigma_{ii_1}\sigma_{ij_1}} + w_2\frac{\sigma_{ij_2}}{\sigma_{ii_2}\sigma_{jj_2}} \quad \text{for} \quad i \neq j. \end{aligned} \tag{3.23}$$

These conditions on the elements of the covariance matrices gave approximately from 30% to 70% of positive definite matrices when attempting a split move.

For the reverse splitting move, random numbers are generated independently from the following distributions: $u \sim be(2,2)$, $v_i \sim N(0,5/8)$, $t_{ii} \sim be(1,1)$ for $i = 1,\ldots,p$, $t_{ij} \sim N(1/2,5/8)$ for $i = 1,\ldots,p \quad i > j$. This random vector is centered using Brooks *et al* [11], so called *"weakly non-identifiability centering"*.

Many applications rely on empirical tuning of the jump functions and the proposal distribution. Brooks *et al* [11] discussed various mechanisms for guiding the choice of the proposal distribution, particularly to center and scale it. Using weakly non-identifiability centering to define the proposal distribution, the authors reported some improvement in the acceptance rate for the split/combine move in some of the examples given in Richardson and Green [51]. For example, the acceptance rate for the split/combine move for the acidity data increased from 8% to 10%. For the same data set, the results obtained for the scaling methods had the acceptance rate reduced from 8% to 2.5%. However, in both cases, there was no need of pilot tuning.

Weakly non-identifiability centering finds a point at which the likelihood contributions are the same for models in both dimensions, we have used this point to center the proposal distribution. In this case the use of Brooks *et al* techniques to scale the parameters of the distributions gave a state dependent variance which in most of the cases was too big and easily led to numerical problems. Therefore, we did not scale the distributions with this technique.

The random numbers defined above provide all required weights and parameters that satisfy the conditions given by equations (3.23). When one component j_* is

selected to split into two new ones j_1, j_2 then the values for the parameters for components j_1 and j_2 are given by,

$$\begin{aligned}
w_1 &= uw_*, \qquad (3.24)\\
w_2 &= (1-u)w_*,\\
\mu_{ij_1} &= \mu_{ij_*} - v_i\sqrt{w_2/w_1}\sigma_{ii_{j_*}},\\
\mu_{ij_2} &= \mu_{ij_*} + v_i\sqrt{w_1/w_2}\sigma_{ii_{j_*}},\\
\sigma^2_{ii_{j_1}} &= t_{ii}(w_*/w_1)\sigma^2_{ii_{j_*}},\\
\sigma^2_{ii_{j_2}} &= (1-t_{ii})(w_*/w_2)\sigma^2_{ii_{j_*}},\\
\sigma_{ij_{j_1}} &= t_{ij}(w_*/w_1)^2\sqrt{t_{ii}t_{jj}}\sigma_{ij_{j_*}},\\
\sigma_{ij_{j_2}} &= (1-t_{ij})\sqrt{(1-t_{ii})(1-t_{jj})}(w_*/w_2)^2\sigma_{ij_{j_*}}.
\end{aligned}$$

The probabilities of accepting these moves are obtained from equation (1.20), considering the expression

$$\begin{aligned}
\frac{p(x'|y)}{p(x|y)} &= \frac{p(k+1|\lambda)}{p(k|\lambda)}\frac{p(\mathbf{w}^{(k+1)}|k+1,\delta)}{p(\mathbf{w}^{(k)}|k,\delta)}\\
&\times \frac{p(\mathbf{z}^{(k+1)}|k+1,\mathbf{w}^{(k+1)})}{p(\mathbf{z}^{(k)}|k,\mathbf{w}^{(k)})}\frac{p(\theta|k+1)}{p(\theta|k)}\frac{p(\mathbf{y}|\boldsymbol{\theta}',\mathbf{z}^{(k+1)})}{p(\mathbf{y}|\theta,\mathbf{z}^{(k)})},
\end{aligned}$$

where $\boldsymbol{\theta}' = (\boldsymbol{\mu}_i, \Sigma_i)$ for $i = 1,\ldots,(k+1)$ and $\boldsymbol{\theta} = (\boldsymbol{\mu}_i, \Sigma_i)$ for $i = 1,\ldots,(k)$. The superindices $(k+1)$ and (k) denote the size of the vector for $\mathbf{w}$ and $\mathbf{z}$.

The acceptance probabilities for the split and combine moves have a rather complex form given by $\min\{1, A\}$ and $\min\{1, 1/A\}$ respectively, where A is given by

$$\begin{aligned}
A &= (\text{likelihood ratio})\frac{1}{B(k\delta,\delta)}\frac{w_{j_1}^{n_{j_1}+\delta-1}w_{j_2}^{n_{j_2}+\delta-1}}{w_{j_*}^{n_{j_*}+\delta-1}}(2\pi)^{-p/2}|\kappa^{-1}|^{-1/2} \qquad (3.25)\\
&\times |(2\beta)^{-1}|^{-\alpha}|\Sigma_{j_1}|^{(2\alpha-p-1)/2}|\Sigma_{j_2}|^{(2\alpha-p-1)/2}|\Sigma_{j_*}|^{(p+1-2\alpha)/2}\\
&\times \exp\left\{-\frac{1}{2}\left[(\boldsymbol{\mu}_{j_1}-\xi)^T\kappa(\boldsymbol{\mu}_{j_1}-\xi)+(\boldsymbol{\mu}_{j_2}-\xi)^T\kappa(\boldsymbol{\mu}_{j_2}-\xi)-(\boldsymbol{\mu}_{j_*}-\xi)^T\kappa(\boldsymbol{\mu}_{j_*}-\xi)\right]\right\}\\
&\times \mathbf{K}^{-1}\exp\left\{-\frac{1}{2}\left[tr(2\beta)\Sigma_{j_1}^{-1}+tr(2\beta)\Sigma_{j_2}^{-1}-tr(2\beta)\Sigma_{j_*}^{-1}\right]\right\}\\
&\times \frac{d_{k+1}}{b_k P_{alloc}}\{h(\mathbf{u},\mathbf{v},\mathbf{t})\}^{-1}|\mathbf{J}|,
\end{aligned}$$

where

$$\mathbf{K} = 2^{\alpha p}\pi^{p(p-1)/4}\prod_{s=1}^{p}\Gamma(\frac{2\alpha+1-s}{2}).$$

The first four lines in the above expression correspond to the ratio $\frac{p(x'|y)}{p(x|y)}$ and the last line corresponds to the ratio $\frac{r_m(x')h'_m(u')}{r_m(x)h_m(u)}\left|\frac{\partial(x',u')}{\partial(x,u)}\right|$ where P_{alloc} is the probability that particular allocation is made and $h(\mathbf{u},\mathbf{v},\mathbf{t})$, denotes the joint density of all the random numbers generated to make the move. The Jacobian for the transformation $|\mathbf{J}|$, was explicitly computed up to the three dimensional case. As dimension increases, the matrix shows the same form in its elements[1]. Hence, we conjecture it has a general expression given by

$$|\mathbf{J}| = \frac{w_{j_*}\prod_{i=1}^{p}(\sigma^3_{ii_{j_*}}(t_{ii}(1-t_{ii}))^{(p-1)/2})\prod_{i=1}^{p}\prod_{j=i+1}^{p}\sigma_{ij_{j_*}}}{(u(1-u))^{(2p^2+p)/2}}.$$

(vi) The birth of a new empty component or the death of an empty component.

This move is done practically in the same way as for the one-dimensional case.

In this case, for the birth move, the parameters of the new component are drawn independently from the following distributions

$$\begin{aligned} w_{j_*} &\sim be(1,k), \\ \boldsymbol{\mu}_{j_*} &\sim N_p(\xi,\kappa^{-1}), \\ \Sigma_{j_*} &\sim W_p(2\alpha,(2\beta)^{-1}). \end{aligned}$$

The weights of the existing components must also be rescaled to satisfy the restriction that the weights of all components should add up to one. The acceptance probability for the birth move is $\min\{1,A\}$ where

$$A = \frac{(1-w_{j_*})^{n+k\delta-k}w_{j_*}^{\delta-1}}{B(k\delta,\delta)}\frac{d_{k+1}}{b_k k(k_0+1)}, \tag{3.26}$$

where k_0 is the number of empty components in the current state.

For the death move, a random choice from the existing empty components is made and the weights of the remaining components are rescaled. The move is accepted with probability $\min\{1,1/A\}$ with A as in the birth move.

[1]See Appendix A for details on the computations.

3.4.1 Examples

In this section we inspect the performance of the method and our implementation in higher dimensions. Using the hierarchical model described in the previous section, a reversible jump sampler was used to fit a mixture of multivariate normal distributions to several data sets[2]. We will follow five examples of different dimensions throughout this work. The examples include data which belongs to well separated clusters as well as typical examples used to illustrate robust and fuzzy clustering. The RJMCMC samplers were run for 300000 iterations, a burn-in of 200000 iterations followed by 100000 iterations, thinned every 50. The resulting 2000 iterations were used for the inference we present in the following sections. The length of the runs (burn-in and monitored iterations) is believed to be enough to give meaningful results, further discussion will be given in Chapter 6.

Example 1

The first example is taken form McLachlan and Peel [43], who considered a three dimensional mixture given by

$$f(x|\theta_j) = 0.66\, N_2(\mu_1, \Sigma_1) + 0.17\, N_2(\mu_2, \Sigma_2) + 0.17\, N_3(\mu_3, \Sigma_3);$$

where the values for the corresponding mean vectors and covariance matrices are the following:

$$\mu_1 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \quad \Sigma_1 = \begin{pmatrix} 9 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

$$\mu_2 = \begin{pmatrix} -6 \\ 3 \\ 6 \end{pmatrix}, \Sigma_2 = \begin{pmatrix} 4 & -3.2 & -0.2 \\ -3.2 & 4 & 0 \\ -0.2 & 0 & 1 \end{pmatrix},$$

$$\mu_3 = \begin{pmatrix} 6 \\ 6 \\ 4 \end{pmatrix}, \quad \Sigma_3 = \begin{pmatrix} 4 & 3.2 & 2.8 \\ 3.2 & 4 & 2.4 \\ 2.8 & 2.4 & 2 \end{pmatrix}.$$

[2]We are grateful to Dr. William Browne, University of Nottingham, who kindly provided us with the C code for the random generators of Wishart and multivariate normal distributions.

A sample of 300 observations was generated, 200 observations were simulated from the first normal distribution, 52 and 48 observations were simulated from the other two respectively.

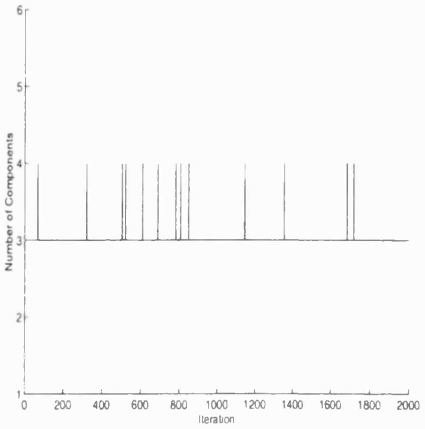


Figure 3.1: Sampled mean values for the number of components by iteration: Simulated data from Example 1.

As shown in Figure 3.1, the number of components sampled in this example was either 3 or 4 and when the sampler accepted 4 components, the fourth one was always empty. The mixing of the sampler is not good in terms of the number of components. It suggests that well separated multivariate normal distributions are easily identified by the sampler.

With respect to the parameters θ_i, sampled values suggest that posterior estimates should be adequate. The sampled means for all three components are shown in Figure 3.2. There was no label switching in this case. The dashed lines correspond to the sample mean values obtained from the simulated data. For each component the sampled values are centered somewhere near the values obtained from the data. The histograms of the covariance matrix elements are shown in Figure 3.3. The dashed lines correspond to the values obtained from the sample covariance matrix of the simulated data. Although there is a suspicion of some bias for some elements of the covariance matrices, overall inspection of the estimates of generalised variance of the components reveals a slight tendency to positive skewness and little evidence of bias, except possibly in the third component, see Figure 3.4. The weights for the 3-component normal mixture are given in Figure 3.5 and correspond well to the values given by the simulated data.

The convergence of this sampler is not easy to assess, following Richardson and Green [51] and Stephens [59], we concentrate for the moment on evaluating the ability

of the RJMCMC sampler to move between different values of k. It is important to give some remarks about the mixing of the sampler over the number of components, k. This parameter is discrete and we expect the sampler to explore several values of k for iterations at the beginning of the simulation. However, once convergence is achieved the number of components tends to stabilize and the resulting output will frequently show a poor mixing for k. We do not consider this as a negative aspect as we expect the sampler to find an adequate number of components to describe the structure of the data exhibiting small variations around this number. As a consequence periods of constant values of k will often be observed.

Once we have considered the mixing over the number of components, k, we looked at the autocorrelation functions and the ergodic averages of the individual parameters, conditional on the number of components. We found there is no evidence that suggests the sampler has not yet reached equilibrium. The sampled values mixed well in the case of the elements of the mean vectors and covariance matrices. Various initial states were used, and similar answers were obtained. We will defer further discussion of the convergence assessment to Chapter 6.

In relation to the acceptance rates for the moves, for the split/combine move only 0.1% of the moves were accepted, being all combine type moves. For the birth/death move 1% of the moves were accepted.

One of the main objectives of any cluster analysis is the partitioning of the data into clusters. Following a suggestion of O'Hagan in the discussion of Richardson and Green [51], one method of achieving this is to calculate a dissimilarity between all pairs of observations derived from counting the number of times each pair is classified in the same component. This may then be used in one of the plethora of available clustering techniques described in Chapter 1.

In this case, a hierarchical clustering was carried out for several distance criteria. Results are robust in that they did not show much variation in the obtained classification. The number of observations allocated in each component was 200, 52 and 48 respectively, which were the number of observations simulated originally sampled from each component.

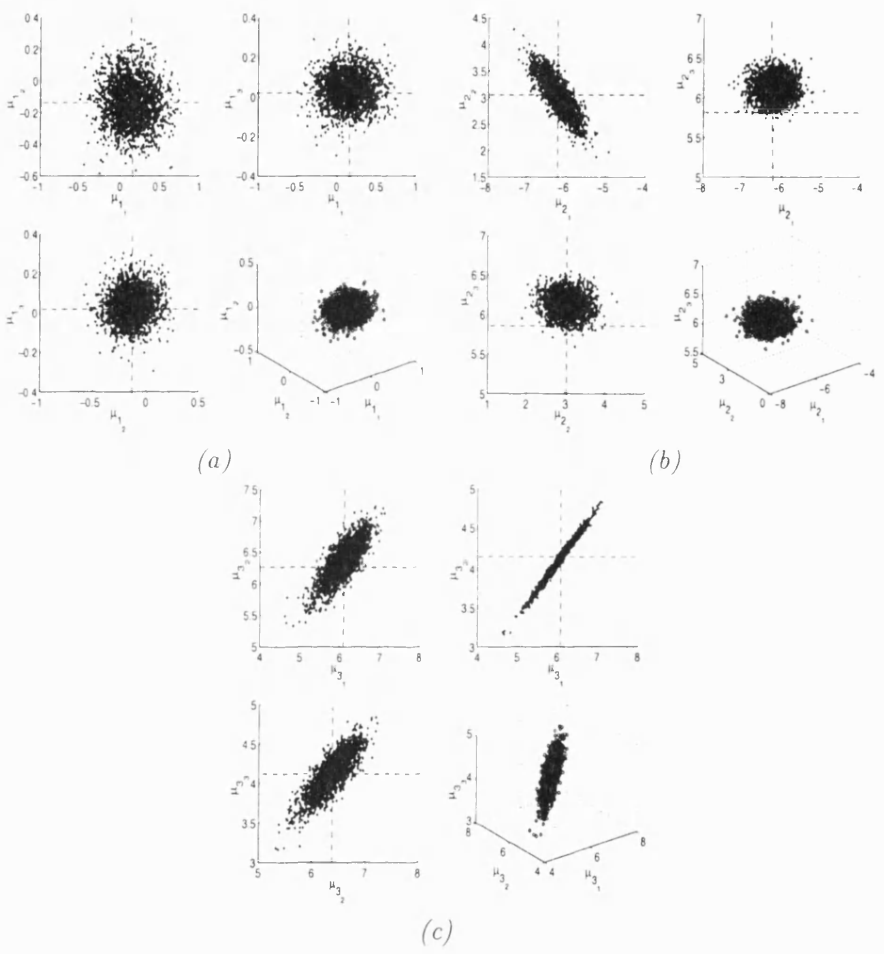


Figure 3.2: Sampled mean values for the 3-component normal mixture: Simulated data. The dashed lines correspond to the values obtained from the simulated data. (a) First Component. (b) Second Component. (c) Third Component.

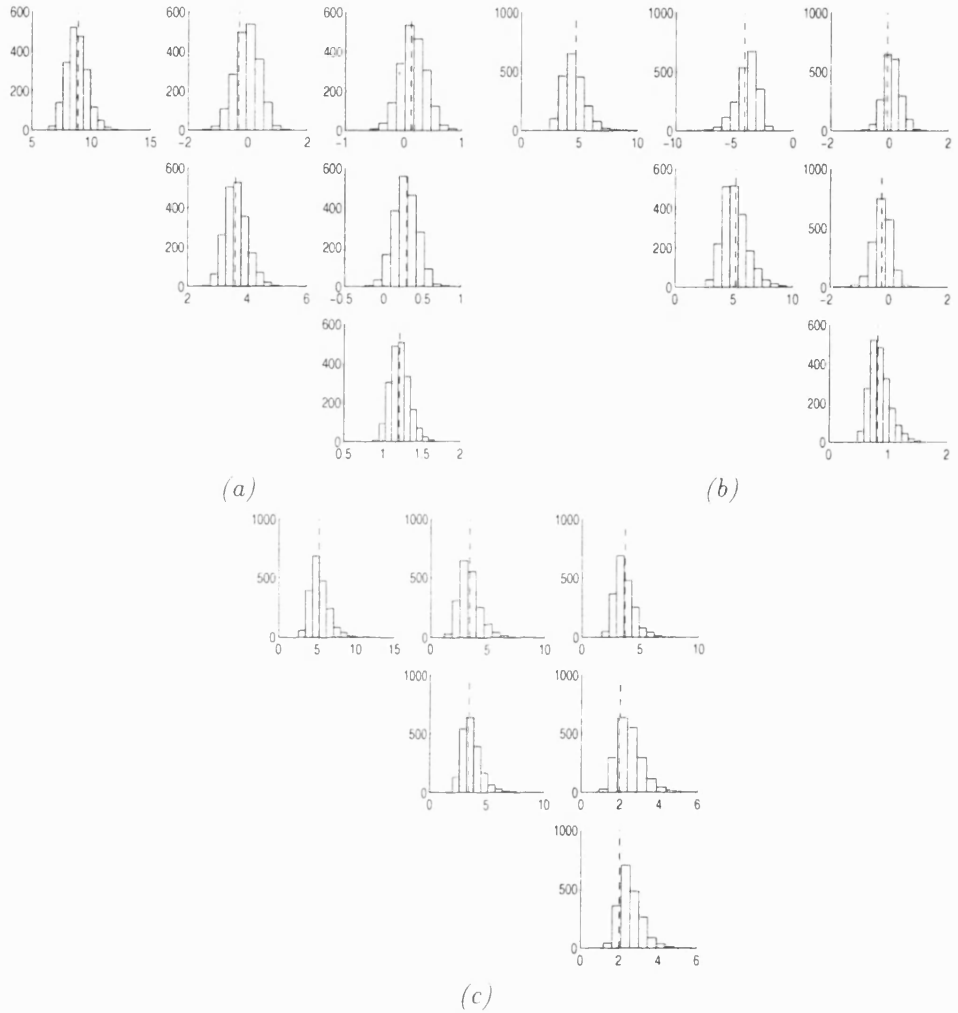


Figure 3.3: Histograms for the sampled covariance matrices for the 3-component normal mixture: Simulated data. The dashed lines correspond to the values obtained from the data. *(a)* First Component. *(b)* Second Component. *(c)* Third Component.

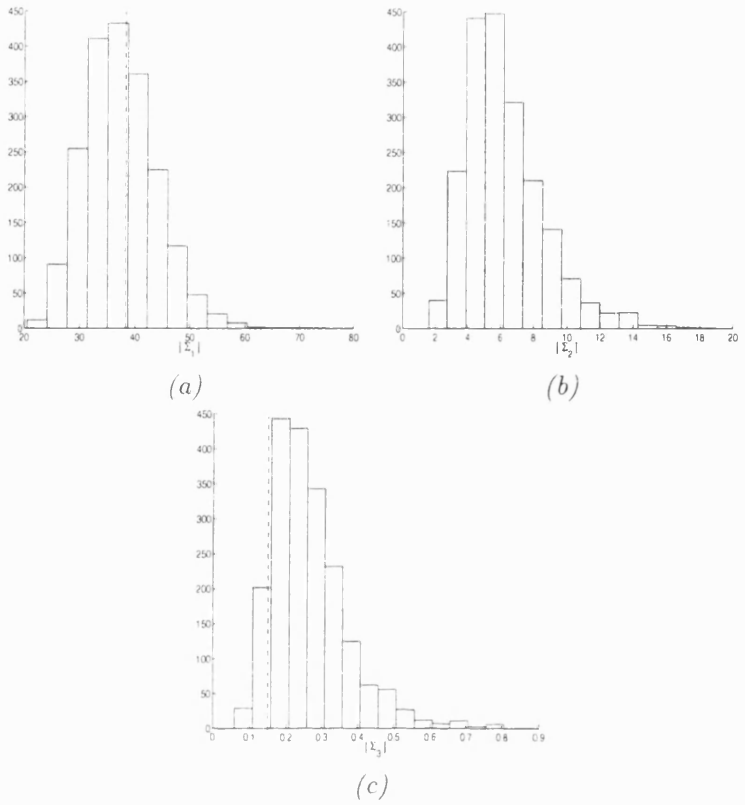


Figure 3.4: Histograms for the sampled generalised variance for the 3-component normal mixture: Simulated data. The dashed lines correspond to the values obtained from the data. *(a)* First Component. *(b)* Second Component. *(c)* Third Component.

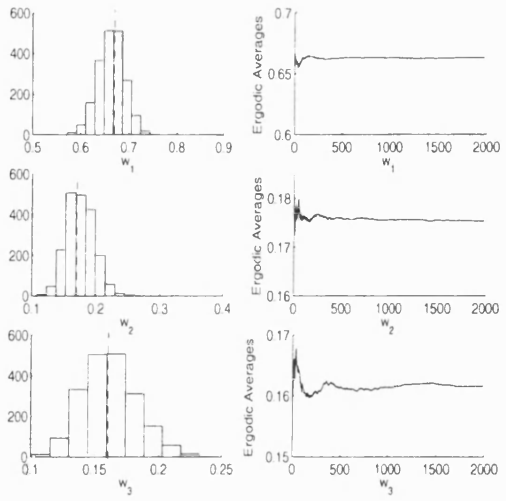


Figure 3.5: Histograms and ergodic averages for the sampled weights of the 3-component normal mixture: Simulated Data.

Example 2

The second example is the data on duration and waiting time before the next eruption from 272 eruptions of the Old Faithful geyser in Yellowstone National Park (the data version used in Härdle [36], Venables and Ripley [64] and Stephens [59]). Each observation records in the first feature the duration of the eruption and in the second feature the waiting time before the next eruption. Both measurements are made in minutes and they are shown in Figure 3.6.

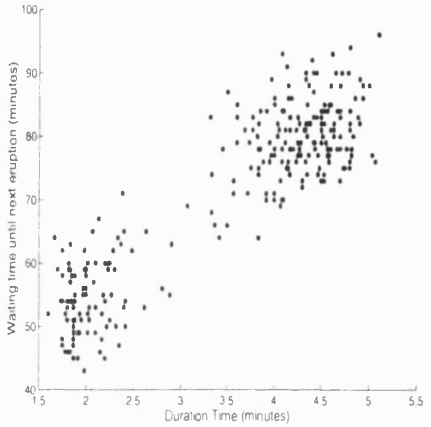


Figure 3.6: Old Faithful data.

A reversible jump sampler was run for this data set, from the inspection of the plot shown in Figure 3.6, we expected the sampler to identify two groups in the Old Faithful data. However, the output shows the data are described by a two or three-component mixture with high posterior probability, those in red corresponding to empty components. Considering that when three components are fitted, there is one small weighted component, this could suggest a component is being used to explain departures from multivariate normality. Therefore, the values that are given the highest posterior probability might not correspond to the number of groups that best describe the data. To decide on the number of groups other aspects, such as the weights of the components and the allocated observations, should be taken into account.

The mixing over the number of components is slightly better for this data set, see Figure 3.7.

The sampled values for the mean vectors of the three-component mixture are shown in Figure 3.8(a). Label switching is clearly observed in this example. If the data are classified using the dissimilarity matrix, it would separate observations 6, 84, 33, 131,

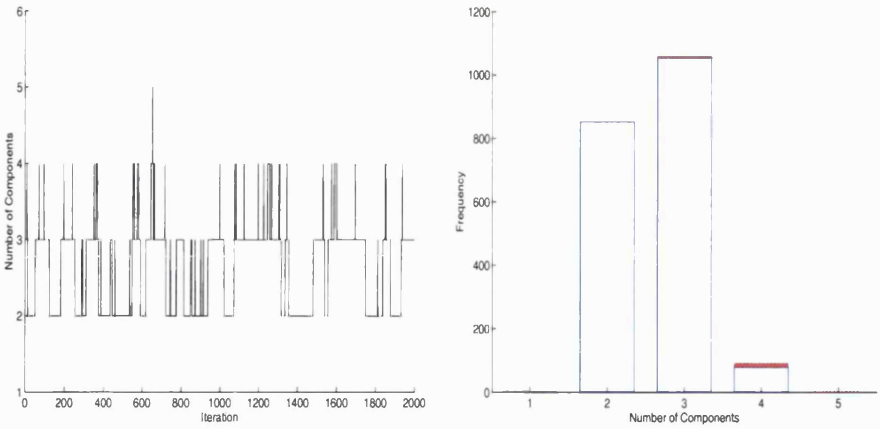
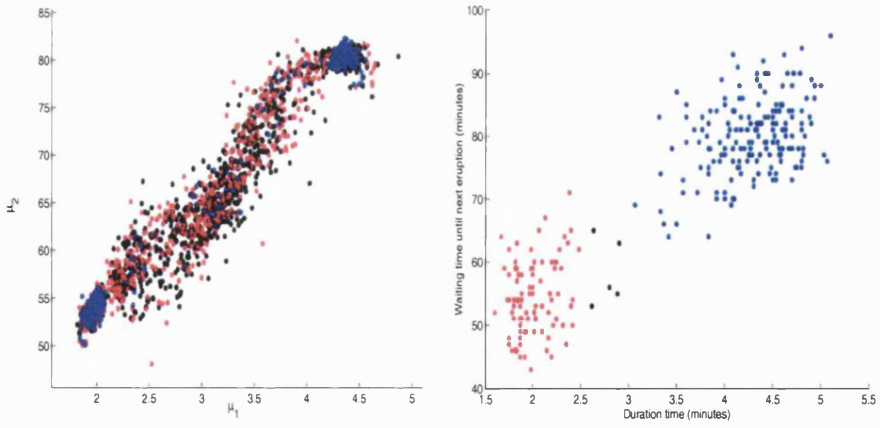


Figure 3.7: Sampled values for the number of components k by iteration and barplot of the number of components k (empty components in red): Old Faithful data.

133 and 244 into a third cluster (black). This favours the possibility that the third component is a consequence of non-normality rather than a separate cluster. If they were allocated into two groups only, these observations would be placed into the red component, Figure 3.8(b), that has 92 observations allocated into it. The remaining 175 observations are classified into the remaining cluster (blue).



(a) *(b)*

Figure 3.8: *(a)* Sampled values for the mean vectors for the three-component mixture: Old Faithful data. *(b)* Classification for the Old Faithful data into three groups.

Once again the convergence analysis (not shown) of the individual monitored parameters, conditional on the number of components, shows no evidence that suggests the chain has not yet reached equilibrium. The acceptance rates for the split/combine move were 0.12% and for the birth/death move were 1%, the same as in the previous example.

Example 3

The third example is the well known artificial Ruspini data [54], [55], which are shown in Figure (3.9). This data set is often used as an example in fuzzy and robust cluster analysis.

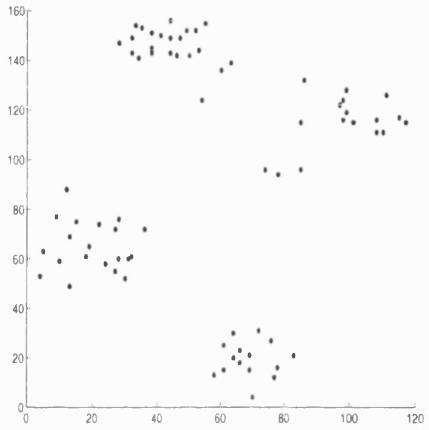


Figure 3.9: Ruspini data.

The fact that small weighted components often reflect departures from normality becomes evident with this data set. Here, a six component mixture is fitted with the highest posterior probability, as can be seen in Figure 3.11. This suggests that the two small components fitted close to the upper groups could be used to cope with non-normality.

The results remain very similar to previous examples, in that mixtures with a larger number of components include several empty ones. The sampler mixes well over the number of components, see Figure 3.10, and the rest of the convergence monitoring gives no evidence suggesting that the chain has not stabilised. The acceptance rate for the split/combine move is 6.3% and for the birth/death move is 12.1%, a lot higher than in previous examples.

The classification for this data set is shown in Figure 3.12. Here we noticed results vary slightly when changing agglomeration criterion used in the hierarchical clustering. Specifically, with the single linkage criterion, observation 45 is separated from the black group to become a different group. With the other criteria, it is observation 41 that is separated from the red group. The latter could suggest that potential outliers could also be separated using small weighted components. We shall revisit this example in the following sections, we fully recognise the limitations of using only normal components.

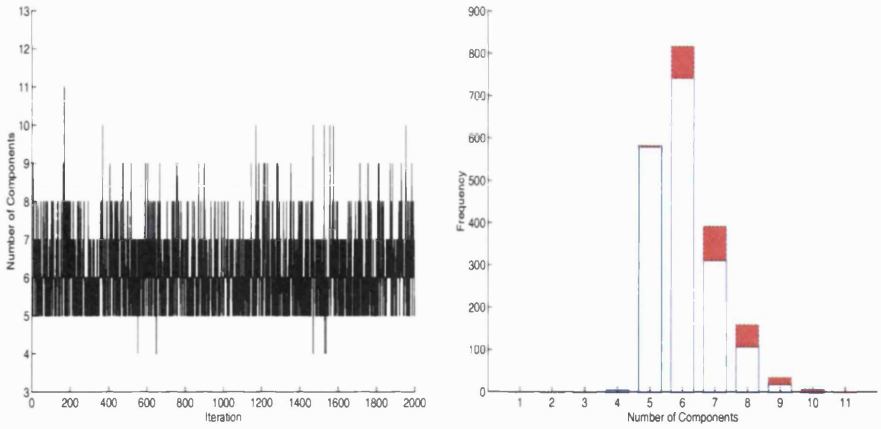


Figure 3.10: Sampled values for the number of components k by iteration and barplot of the number of components k: Ruspini data.

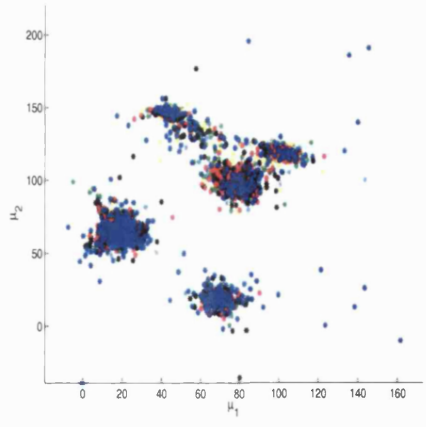


Figure 3.11: Sampled values for the mean vectors for the six-component mixture: Ruspini data.

This will be further discussed in chapter 7.

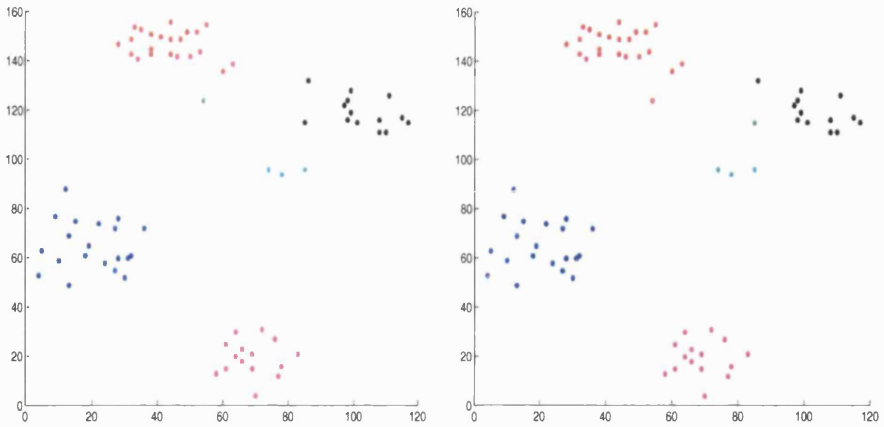


Figure 3.12: Classification for the Ruspini data into six groups. *(a)* Single linkage. *(b)* Complete and average linkage.

Example 4

The next example is the Iris data collected by Anderson [1] in 1935. He recorded four measurements, petal and sepal length and width for 50 specimens of each of three species of iris (*setosa*, *versicolor* and *virginica*).

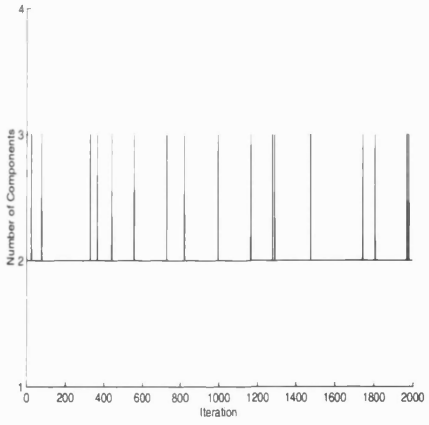


Figure 3.13: Sampled values for the number of components k by iteration: Iris data.

In this case the number of components showed a bad mixing, see Figure 3.13, keeping a two-component mixture throughout most of the monitored period. The sampler is not able to distinguish between the first two species of iris and placed them together in one group. The mean vectors for these two species are very similar for certain elements and clearly different for some others, but this group of moves is not useful to separate

them.

As dimension increases the number of accepted split/combine moves is reduced, mainly because the number of positive definite matrices obtained through the moment matching is also reduced. As a consequence, no split moves were accepted and only 847 combine moves were accepted. The birth/death move was accepted in 1% of the iterations.

If the observations were classified using these results the first two species would be put together in one group and the remaining 50 observations would be allocated into a second group.

Example 5

The last example corresponds to the Lubischew's [42] Beetle data from 1962. These data consist of 74 specimens from three different species of male flea-beetles of the genus *Chaetocnema*, (*concinna*, *heikertingeri* and *heptapotamica*). The features measured were:

1. width of the first joint of the first tarsus, in microns (sum for both tarsi),
2. width of the second joint of the first tarsus, in microns (sum for both tarsi),
3. the maximal width of the head between the external edges of the eyes, in units of 0.001mm,
4. the maximal width of the aedeagus in the fore part, in microns,
5. the form angle of the adageous, in units of 7.5^o,
6. the aedeagus width in microns.

For this example, when considering all the six variables, the reversible jump sampler fitted only two components, see Figure 3.14. It is not possible for the sampler to distinguish the first and the last species of beetle. When classifying the observations through the dissimilarity matrix, observations from the first and last group are placed together. The mixing over the number of components is not very good, see Figure 3.14 (a). When the sampler fitted a three-component mixture there was always one empty component. Here, no split/combine moves were accepted and for the birth/death move the acceptance rate was of 3.1%.

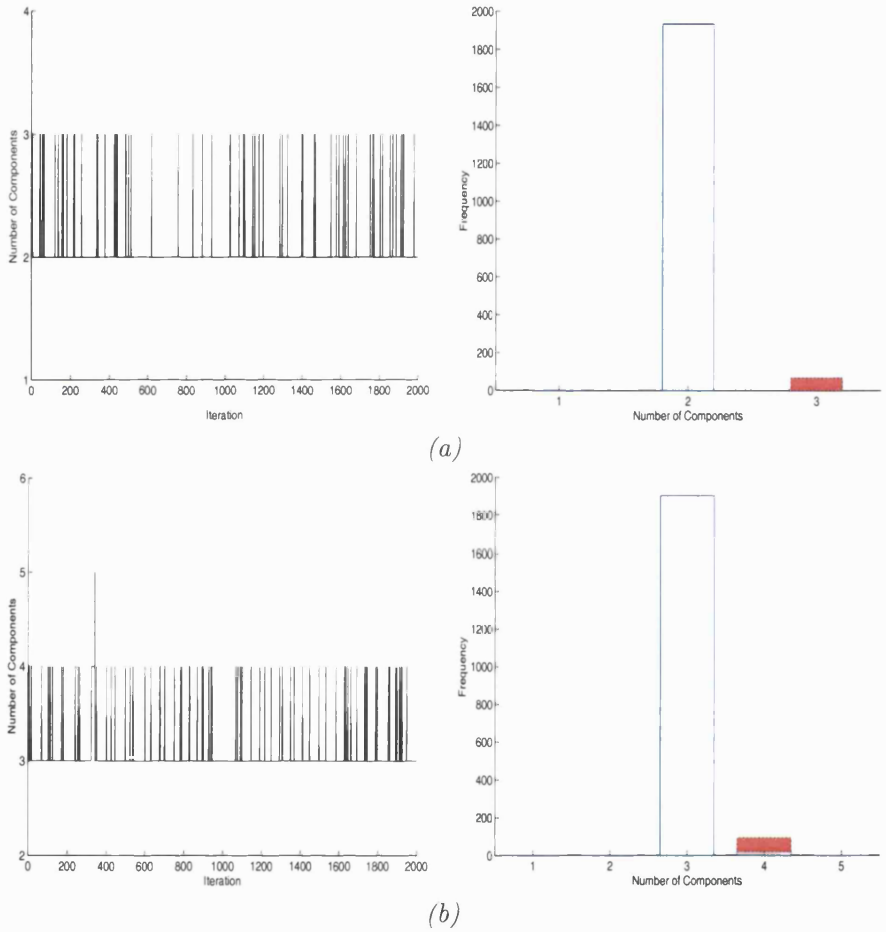


Figure 3.14: Sampled values for the number of components k by iteration and barplot of the number of components k: Lubischew's Beetle data. *(a)* Considering six variables. *(b)* Considering five variables.

We then excluded the fifth variable considering that there is a difference in scale and a relatively discrete set of values. In this case, the sampler fits a three-component mixture. For the 93 iterations where a fourth component is included, it was empty in 75. The mixing over the number of components is not good, see Figure 3.14 (b), but the sampled values for the mean vectors and covariance matrices are very close to the values obtained from the data. When classifying the observations they are adequately separated into the three species. For this example the acceptance rate for the birth/death move is 3.4% but only one split/combine move was accepted.

3.4.2 Sensitivity analysis for posterior inferences.

Richardson and Green [51] and Stephens [59] have discussed that the posterior distribution for the number of components k depends not only on the hyperprior specifications for the distribution of k, but also on the priors given to other parameters of the mixture.

We have assumed on the previous examples a uniform prior on $\{1, \ldots, k_{\max}\}$ for k. A Poisson distribution was considered for the Ruspini data set, with $\lambda = 5, 10$, see Figure 3.15.

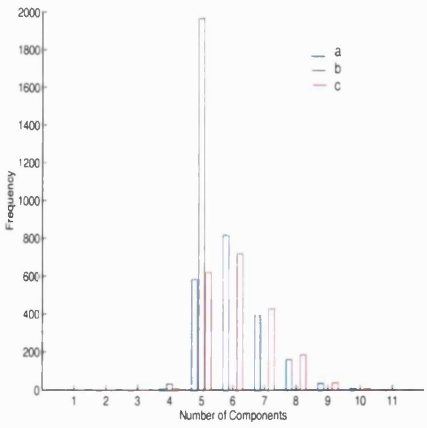


Figure 3.15: Posterior number of components k: Ruspini data. (a) Uniform prior on $\{1, \ldots, k_{max}\}$. (b) Poisson prior with $\lambda = 5$. (c) Poisson prior with $\lambda = 10$.

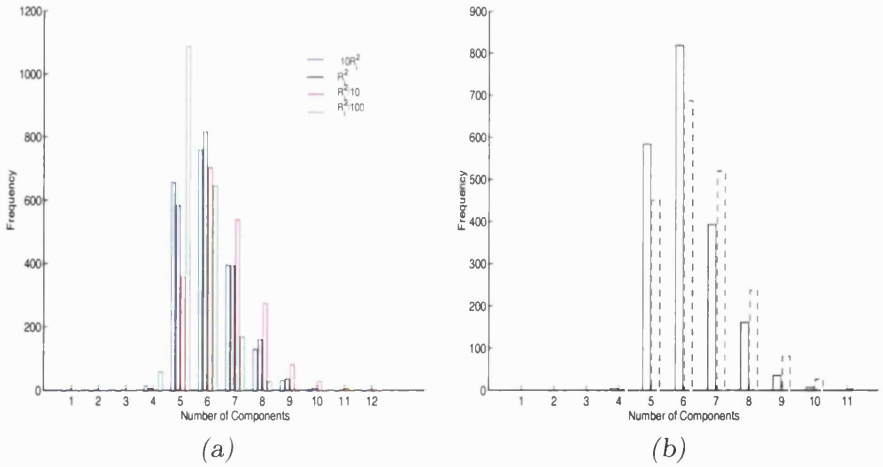
When $\lambda = 5$, the distribution for the posterior number of components has its mode at five components. We noticed the acceptance rates for both move types, split/combine and birth/death, were reduced to approximately 0.1%. The sampled values for the mean vectors show that one component is given a larger variance and therefore, two groups are represented by only one in this case. For $\lambda = 10$, the sampler gives very

similar results to those obtained when considering a uniform prior distribution for the number of components. A six-component mixture is given a larger probability in the posterior distribution for the number of components and the remaining parameters show similar values. Perhaps encouraging a larger number of components through the prior will only give further information if the data are supporting the existence of the given prior modal value.

Varying the prior on the covariance matrices, once it is ensured that the conditional posterior will be proper, did not show important differences for the posterior distribution of k when different values of α, g and h were considered.

The priors on the means $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k$ were taken as multivariate normal, $N_p(\xi, \kappa^{-1})$, for a fixed value of ξ and κ^{-1}. Richardson and Green reported a subtle change in the posterior number of components for different values of κ^{-1} in the one dimensional case. They showed that when the values of κ^{-1} were reduced, the number of components with the highest posterior probability first increased to reach a peak and then decreased again.

We observe a similar behaviour in the multivariate case, but changes in the values are even more subtle here. We considered values from $10R_i^2$ to $R_i^2/100$ in each entry of the diagonal inverse covariance, where R_i is the range of the observed values. In the Ruspini data, the number of components with highest posterior probability is six in almost every case, until it is reduced again for $R_i^2/100$, see Figure 3.16 (a).



(a) (b)

Figure 3.16: (a) Posterior number of components for different values of the covariance matrix in the Ruspini data. (b) Solid line: posterior number of components for a model with κ fixed with R_i^2 in the diagonal entries. Dashed line: number of components for the model with ξ and κ taken as hyperparameters. Ruspini data.

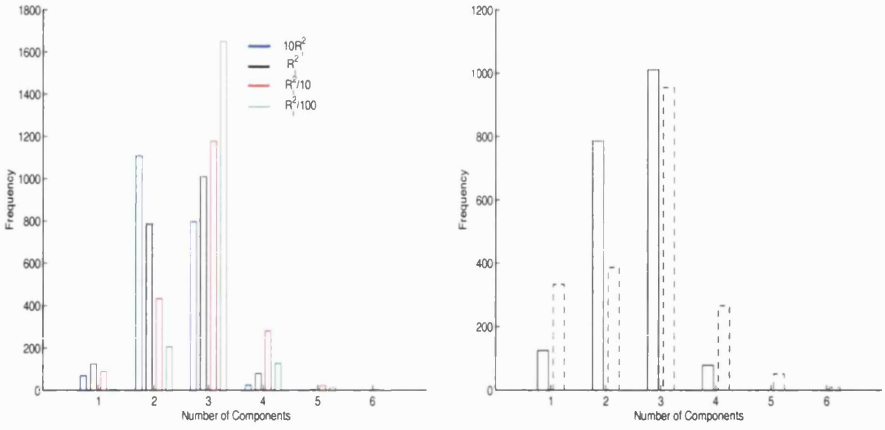
To address this situation, Stephens considered a " vague " prior on the hyperparameters ξ and κ. An improper uniform prior was given to ξ and a " vague " $W_p(l, (lI)^{-1})$ for κ. The latter distribution is fixed to be proper by considering $l = p - 1 + \epsilon$. He pointed out that fixing priors to be proper like this is not a good thing to do, but he goes on explaining that for this l, inference for μ, Σ and k is not sensitive for small values of ϵ. However, numerical problems easily occur for small values of ϵ, we consider $\epsilon = 0.5$ instead. The posterior full conditionals for ξ and κ are

$$\begin{aligned} \xi|\cdots &\sim N_p(\overline{\mu}, (k\kappa)^{-1}) \\ \kappa|\cdots &\sim W_p(l+k, (lI_p + SS)^{-1}), \end{aligned} \tag{3.27}$$

where $\overline{\mu} = \sum_i \mu_i/k$ and $SS = \sum_i (\mu_i - \xi)(\mu_i - \xi)^T$.

The conclusions drawn from the posterior distribution for the number of components do not differ greatly from the ones obtained by keeping κ as a fixed value, see Figure 3.16 (b). The posterior values sampled for the remaining parameters remain unchanged for several values of ϵ. Notice that very small values for ϵ were not considered as they lead to numerical problems. Comparing the sampled values for the component parameters when κ is fixed and variable, in the latter case one of the components is given a larger variance and the sampled values for the mean vector are more dispersed. However, if the data were classified using the dissimilarity matrix described in the previous section, the classification remains unaltered.

The analysis showed very similar results for the Old Faithful data set, see Figure 3.17. The number of components with the highest posterior probability increased as the values in the diagonal of the inverse covariance matrix are reduced from the first value considered, $10R_i^2$. A three-component mixture has the highest posterior probability for the models with κ fixed and R_i^2, $R_i^2/10$ and $R_i^2/100$ in the diagonal entries. The number of components with the highest posterior probability was not reduced for the inspected values of κ. In this case the remaining parameters are very similar in all the settings and the classification using the dissimilarity matrix obtained from the sampler with variable κ was the same as the one obtained with the fixed κ.



(a) (b)

Figure 3.17: (a) Posterior density estimator for the number of components for different values of the covariance matrix in the Old Faithful data. (b) Solid line: posterior density for the number of components for a model with κ fixed with R_i^2 in the diagonal entries. Dashed line: posterior density for the model with ξ and κ taken as hyperparameters. Old Faithful data.

The posterior number of components is highly dependent on the prior assumptions taken, but the assumptions we have taken for the examples to explore the performance of the reversible jump sampler led to very similar conclusions in terms of clustering and classification.

3.4.3 Discussion.

The use of RJMCMC samplers to fit a multivariate normal mixture model presents some challenges. The method is computationally demanding, optimising the program might be necessary for some applications, particularly where time invested to obtain results is an issue. Another crucial aspect is the definition of a family of moves that enables the sampler to change dimension. The performance of the RJMCMC sampler using the split/combine and birth/death moves defined in this chapter was in general efficient. However, posterior inference is not an easy task since problems such as label switching, convergence assessment and sensitivity to prior assumptions always need addressing.

We have seen in the previous section that the posterior distribution of the number of components shows a high dependence on the prior assumptions made on both the number of components and the mean vectors. If there is any information on the number of components or the mean values, this should be incorporated in the prior beliefs.

In terms of density estimation, Bayes factors, given as

$$B_{k_1,k_2} = \frac{p(k_1|y)/p(k_2|y)}{p(k_1)/p(k_2)},$$

could be used to compare models. Theoretically they do not depend on the prior $p(k)$. For example, using the MCMC estimates we compared the models with six and five components for the Ruspini example with a uniform and a Poisson prior: $B_{6,5} = 1.4015$ for a uniform prior with $k_{max} = 30$ and $B_{6,5} = 0.9956$ for a Poisson prior with $\lambda = 10$. When a density estimation problem is considered, a five-component mixture could be proposed as the model since the Bayes factors are close to one. The latter factors could be used to select a good model, other aspects such as parsimony could be added to the selection criteria, as well as the comparison with mixtures of other multivariate densities. However, when considering the clustering problem we need to consider a six-component mixture and compare the results with other models and other estimation methods. That is fitting a six-component mixture does not necessarily mean that there are six groups in the observed data.

In terms of clustering and classification, we found other aspects that must be kept in mind. Overlapped clusters are difficult to separate and departures from normality as well as outliers will often result in fitting more components than *"groups"* in the observed data. Therefore, distinguishing between these two categories might be useful and will be considered Chapter 7. If variable selection is used, it would be important to identify the variables that best discriminate the groups, particularly if they overlap. In general, when the groups are well separated the RJMCMC sampler will fit a model that effectively identifies these groups and additionally it will provide a posterior probability for such a model. These results together with other alternatives to fit a multivariate normal mixture model will result in a robust description of the groups in the observed data.

CHAPTER 4

Other split/combine moves for the RJMCMC

4.1 Discussion of RJMCMC approach to multivariate clustering

The split move based on the moment matching type condition has shown to be efficient, particularly when there are well separated groups. However, we notice that the acceptance rate for the split/combine move is considerably smaller compared to the acceptance rate for the birth/death move. In order to increase the acceptance rate for the split/combine move, we allowed the data to give information on where the new parameters could be placed.

We hope that data driven moves help the sampler to identify important gaps or differences in the allocated data of the selected component to split. We propose the use of both univariate projections, particularly in the direction of the principal components, and the minimum spanning tree. Both concepts are often used to guide clustering algorithms as we have seen in Chapter 2.

4.2 Data informed moves based on principal components

Low dimensional projections of multivariate data have been used to provide interesting views of the full-dimensional data, see for example Peña and Prieto [47] . They have looked for gaps that suggest the existence of two modes in one cluster. If there is a gap that separates two clouds of observations that are allocated in one normal com-

ponent that we intend to split into two different ones, this gap could be detected when projecting the data points onto the principal axis of the fitted normal. For illustration see Figure 4.1. Using this idea, a different split/combine move was constructed.

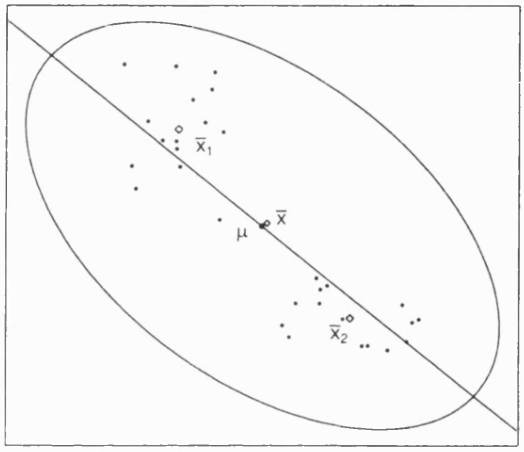


Figure 4.1: An example of a component selected to split, with mean parameter μ. The current sample mean is $\bar{x}$ and when projecting it onto the principal axis it splits the allocated data in two groups with sample means $\bar{x}_1$ and $\bar{x}_2$.

The construction aims to allow data to give information on where the new parameters could be placed when splitting a component into two, so that the move is more likely to be accepted. Using the projected data to determine the parameter values can result in an increase in the likelihood.

First a nonempty component labelled j_* is selected with equal probability from the k existing ones, since any of them should be selected to split. Then we compute the value of the sample mean and the sample covariance matrix, denoted $\bar{\mathbf{y}}_{j_*}$ and S_{j_*} respectively, based on the n_{j_*} observations currently allocated to component j_*. When these are available, we obtain the principal axis from decomposing S_{j_*} into its singular values.

Let $\lambda_1, \ldots, \lambda_p$ be the eigenvalues of S_{j_*}, satisfying $\lambda_1 > \lambda_2 > \cdots > \lambda_p > 0$. Let $\mathbf{t}_{(1)}, \ldots, \mathbf{t}_{(p)}$ be the corresponding eigenvectors with unit length and $T = (\mathbf{t}_{(1)}, \ldots, \mathbf{t}_{(p)})$,

such that

$$T'S_{j_*}T = \text{diag}(\lambda_1, \ldots, \lambda_p).$$

Consider $\overline{\mathbf{y}}'_{j_*}$, the orthogonal projection of $\overline{\mathbf{y}}_{j_*}$ onto $t_{(1)}$. Using $\overline{\mathbf{y}}'_{j_*}$ and the projection of all observations currently allocated to component j_*, $y'_{\{i:z_i=j_*\}}$, into the principal axis, we split the allocated observations in two groups j_1 and j_2. One with those observations whose projection have a smaller value than $\overline{\mathbf{y}}'_{j_*}$ and the other with those observations whose projection have a bigger value than $\overline{\mathbf{y}}'_{j_*}$. If both groups are non-empty with n_{j_1},$n_{j_2} \geq 2$, then values for $\overline{\mathbf{y}}_{j_1}$, S_{j_1}, $\overline{\mathbf{y}}_{j_2}$, S_{j_2} are used to define the new parameters for the components labelled j_1 and j_2. The idea we pursue is to preserve the distance observed between the elements of current parameters (μ_{j_*}, Σj_*) and the sample values ($\overline{\mathbf{y}}_{j_*}$, S_{j_*}). The mechanism that allows the sampler to move between dimensions requires the choice of a set of deterministic functions and random numbers whose distributions should also be established. An important aspect to take into account is that the move should be reversible and we need the chosen deterministic functions to ensure this.

We will consider the distance described above in the positive direction, dividing the resulting segment into two segments that add up to the observed distance. Therefore we use the random numbers $u_i, v_i \in [0,1]$ which are generated independently from beta distributions $u_i \sim be(2,2)$, $v_i \sim be(1,1)$ and the set of deterministic functions is as follows:

$$\begin{aligned}
w_{j_1} &= pw_{j_*}, \\
w_{j_2} &= (1-p)w_{j_*}, \quad \text{where} \quad p = \frac{n_{j_1}}{n_{j_*}}, \\
\mu_{i_{j_1}} &= \overline{\mathbf{y}}_{i_1} + |\mu_{i_*} - \overline{\mathbf{y}}_{i_*}|u_i, \\
\mu_{i_{j_2}} &= \overline{\mathbf{y}}_{i_2} + |\mu_{i_*} - \overline{\mathbf{y}}_{i_*}|(1-u_i), \\
\sigma_{il_{j_1}} &= s_{il_{j_1}} + |\sigma_{il_{j_*}} - s_{il_{j_*}}|v_i, \\
\sigma_{il_{j_2}} &= s_{il_{j_2}} + |\sigma_{il_{j_*}} - s_{il_{j_*}}|(1-v_i),
\end{aligned} \tag{4.1}$$

where $i, l = 1, \ldots, p$ and p is the dimension of the observed variables. The absolute values are used in the conditions firstly to ensure that by considering only the positive difference between the component parameters and the sample values, the covariance matrix has positive elements in its diagonal and secondly to ensure that the random

numbers for the reversible move $u_i', v_i' \in [0,1]$. Although, strictly speaking, using the absolute values does not define a bijection, which is needed to use the RJMCMC, we take into account that in practice only one of the two values for μ_{j_*} and $\sigma_{il_{j_*}}$ that satisfy equations (4.1) will lead the particular allocation of the data into component j_* with higher probability. The function and its inverse must also be differentiable so that the standard change-of-variable formula can be used to specify when the detailed balance condition holds. In order to satisfy this, we exclude the component as a candidate for the PC split move if $\mu_{i_*} = \overline{\mathbf{y}}_{i_*}$ or $\sigma_{il_{j_*}} = s_{il_{j_*}}$, for any $i, l = 1, \ldots, p$, this situation was not observed in practice.

For the reverse combine move, two nonempty components labelled j_1 and j_2 are selected at random, again with probability proportional to the inverse of the Mahalanobis distance, given by equation (3.21), between every pair of means. Once the two components have been selected, they are merged into a new component labelled j_* so that,

$$\begin{aligned} w_{j_*} &= w_{j_1} + w_{j_2}, \\ |\mu_{i_{j_*}} - \overline{\mathbf{y}}_{i_{j_*}}| &= |\mu_{i_{j_1}} - \overline{\mathbf{y}}_{i_{j_1}}| + |\mu_{i_{j_2}} - \overline{\mathbf{y}}_{i_{j_2}}|, \\ |\sigma_{il_{j_*}} - s_{il_{j_*}}| &= |\sigma_{il_{j_1}} - s_{il_{j_1}}| + |\sigma_{il_{j_2}} - s_{il_{j_2}}|. \end{aligned} \tag{4.2}$$

Here again we exclude the component as a candidate for the PC combine move if $\mu_{i_{j_1}} = \overline{\mathbf{y}}_{i_{j_1}}$, $\mu_{i_{j_2}} = \overline{\mathbf{y}}_{i_{j_2}}$, $\sigma_{il_{j_1}} = s_{il_{j_1}}$ or $\sigma_{il_{j_2}} = s_{il_{j_2}}$, for any $i, l = 1, \ldots, p$. For convenience, we select the collection of values which ensure positive elements in the diagonal of the covariance matrix Σ_{j_*} .

The probabilities for accepting the split/combine move remain as given in equation (3.25), provided we adopt a uniform prior distribution over the number of components k. The corresponding Jacobian for this transformation $|\mathbf{J}|$, was again explicitly computed up to the 3 dimensional case. Following the same arguments as for the moment matching type move, we conjecture the Jacobian has a general expression given by

$$|\mathbf{J}| = \left| w_{j_*} \left(\prod_{i=1}^{p} \operatorname{sgn}(\mu_{i_{j_*}} - \overline{\mathbf{y}}_{i_{j_*}}) \right) \left(\prod_{i=1}^{p} \prod_{l=i+1}^{p} \operatorname{sgn}(\sigma_{il_{j_*}} - s_{il_{j_*}}) \right) \right|. \tag{4.3}$$

We expected this PC split/combine move to give the sampler a better opportunity to identify the different clusters increasing the acceptance rates for the split/combine moves. We will illustrate that this is not always the case through the examples in the next section.

4.2.1 Examples

We will discuss the performance of this move through the examples described in previous chapters. This PC based split/combine move was used in a reversible jump sampler with a burn-in period of 200000 iterations followed by 100000 sweeps, thinned every 50.

Example 1: Simulated 3-component mixture.

The simulated data for a mixture of 3 well separated components introduced in Chapter 3 were described adequately, posterior inference did not show any outstanding change from results obtained with the first split/combine move. The PC based split/combine did not have a larger number of moves accepted compared to the moment matching type move. The mixing over the number of components was not good, once the sampler identified the three simulated components, there were only a few iterations with four components, in which case the fourth component was empty in most cases. The acceptance rate for the split/combine move remained less than 0.1%. The sampled values for the parameters showed a very similar behaviour to that observed with the moment matching type split/combine move. There was no indication of lack of convergence from the analysis on individual parameters. The classification of the data using the dissimilarity matrix placed each observation in the component from which they were simulated.

Example 2: Old Faithful data.

For the Old Faithful data, the sampled values for the number of components spent longer periods in one value with the PC split/combine move and the number of components with the highest posterior probability remained three, see Figure 4.2.

The sampled values for the mean vectors are not very distant from the ones obtained with the moment matching type move, see Figure 4.3(a). In relation to the posterior classification, using the built dissimilarity matrix, only three observations were placed

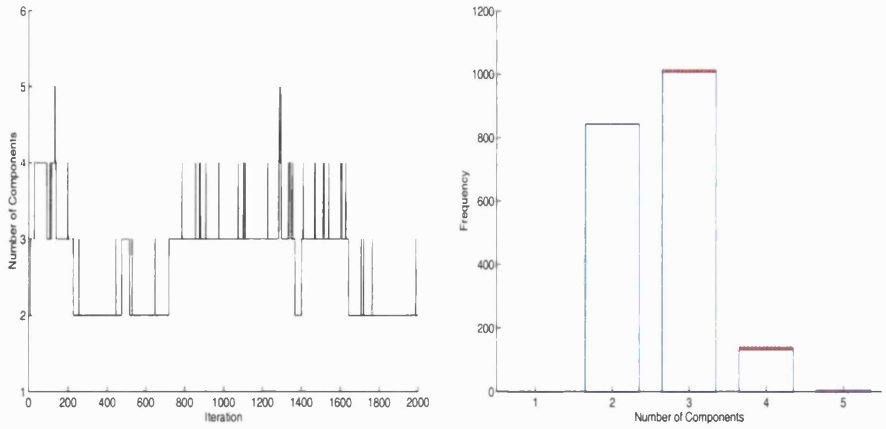
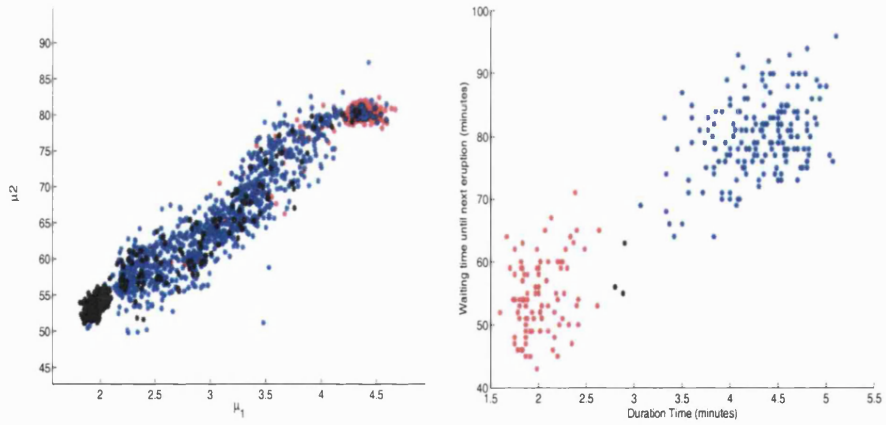


Figure 4.2: Sampled values for the number of components k by iteration and barplot of the number of components k, principal component based split/combine move: Old Faithful data.

on the black group: 6, 133 and 244, Figure 4.3(b). If they were allocated into only two groups these observations would be added to the red group. However, the acceptance rate for the split/combine move is still less than 0.1%.



(a) (b)

Figure 4.3: *(a)* Sampled values for the mean vectors for a three-component mixture, principal component based split/combine move: Old Faithful data. *(b)* Classification for the Old Faithful data into three groups.

Example 3: Ruspini data.

The results obtained for the Ruspini data set were noticeably improved in terms of the acceptance rates: 16.7% for the split/combine move and 23% for the birth/death move. The mixing over the number of components is also good, see Figure 4.4.

Here the number of components with highest posterior probability is nine. The

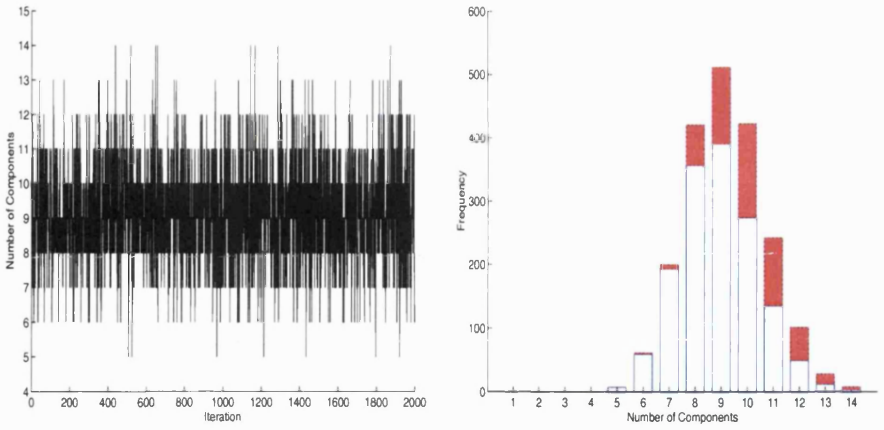


Figure 4.4: Sampled values for the number of components k by iteration and barplot of the number of components k, principal components based split/combine move: Ruspini data.

sampled mean values for the 9-component mixture, excluding iterations with empty components, are shown in Figure 4.5. The corresponding classification based on the dissimilarity matrix obtained from the sampler is also displayed. In this example, the principal axis direction separates the selected component into groups that are not removed because they represent an increase in the likelihood. The latter might also indicate that the presence of nine groups might be reflecting departure from normality of some of the clusters.

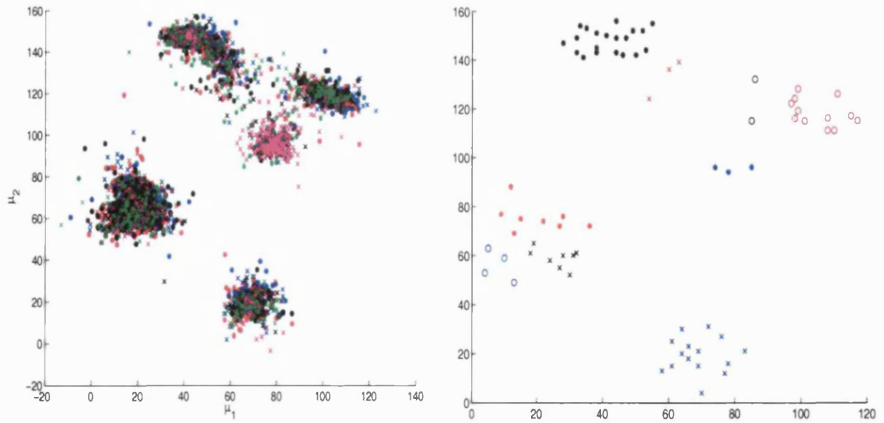


Figure 4.5: (a) Sampled values for the means of a 9-component mixture, principal components based split/combine: Ruspini data. (b) Classification for the Ruspini data set into nine groups.

Example 4: Iris data.

The Iris data set is not successfully explained when using the PC split/combine move. Although this time the sampling rates for both split/combine and birth/death moves were of 1.12% and 1.42%, respectively, compared to the 0.003% and 1% obtained for the moment matching type split/combine move described in Chapter 3. The sampler stayed with a one-component mixture most of the time, sometimes accepting a second one which was always empty, see Figure 4.6.

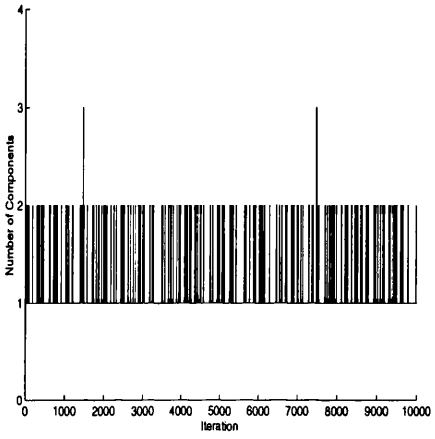


Figure 4.6: Sampled values for the number of components k by iteration, principal components based split/combine move: Iris data.

Example 5: Lubischew's beetle data.

The Lubischew's beetle data were not described with the expected number of clusters when we considered all six variables. In this case no split/combine moves were accepted. The mixing over the number of components is not good and it retains one very wide component and occasionally one empty component is included. However, when the fifth variable was removed from the analysis, the acceptance rate for the split/combine move increased to 2.9% and for the birth/death move 8.3%. The number of components with highest posterior probability is 5. The corresponding classifications using the dissimilarity matrix gives groups of 21, 13, 15, 3 and 22 observations respectively. That is the middle species, the *heikertingeri* is split into three different groups. Once again the the presence of a larger number of components could be related to the fact that compact groups have been found in the direction of the principal axis, and this results in an increased likelihood that remains throughout the sampler. It may

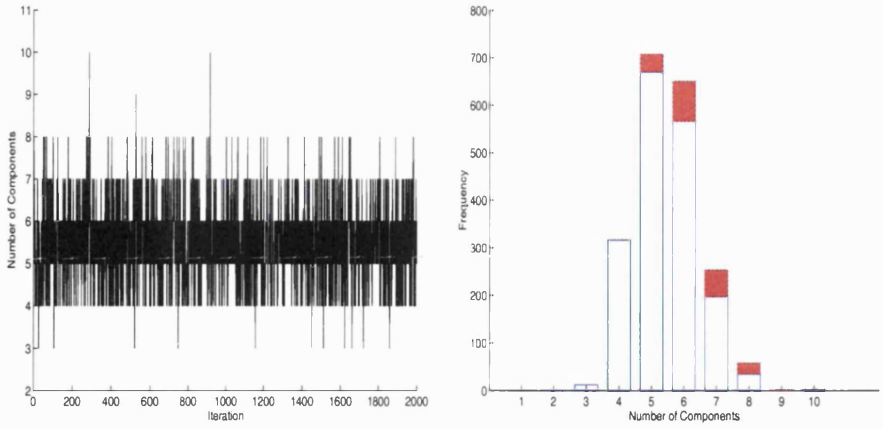


Figure 4.7: (a) Sampled values for the number of components k by iteration and bar plot of the number of components k, principal components based split/combine move: Lubischew's beetle data.

also be reflecting departures from normality.

From these examples, we have seen that the PC split/combine move could help increase the acceptance rates in some cases, particularly when a larger number of components is encouraged by finding compact groups in the direction of the principal axis.

4.3 Data informed moves using minimum spanning trees

When various groups are described by only one component we are interested in a split move that separates different groups, when splitting the selected component into two, in an efficient way. In the previous section we have proposed the use of the projection in the direction of the principal axis to improve the performance of the sampler. We would only expect this move to help when the observations allocated in the component we are splitting show a clear separation when projected onto the principal axis. Here, we will explore the use of a graphical based method to try to identify important gaps between groups, helping the split move to separate groups placed in a single component. This could help the sampler to identify compact groups, within the selected component, in other directions.

As defined by Seber [58]; a *spanning tree* is any set of straight line segments, also known as connected edges, joining various pairs of points, also known as nodes, such that there are no loops. Each point is visited by at least one line, and each point

is connected to every other point either directly or through a chain of intermediary points. The length of the tree is the sum of the lengths of its segments. The *minimum spanning tree* (MST) is the spanning tree of minimum length.

Once a component has been selected to split we look at the minimum spanning tree that connects the observations currently allocated to the component. The lengths of the edges were computed using the Euclidean distance, then we search for the longest edge and remove it, splitting the allocated data into two groups. To determine the parameters of the new components we use again the idea of preserving the distance between the elements of μ_* and $\overline{y}_*$ to define μ_1 and μ_2 using $\overline{y}_1$ and $\overline{y}_2$ as given by equations (4.1-4.2). The acceptance rate remains unchanged and corresponds to equation (3.25) where the jacobian $|\mathbf{J}|$ is given by equation (4.3).

4.3.1 Examples

With this MST split/combine move a reversible jump sampler was run for the examples described in previous sections, with a burn-in period of 200000, monitoring 100000 iterations thinned every 50.

Example 1: Simulated 3-component mixture.

This move is less efficient for the simulated data, no split/combine moves are accepted and the sampler moves only through the birth/death move, accepting 1% of this move type. The sampler mixes poorly over the number of components, keeping most of the time a three-component mixture. The values sampled for the remaining parameters would adequately describe the three different normal distributions used to simulate the data. Results in general match the ones obtained from the reversible jump samplers using the other split/combine moves. The parameters considered individually did not show any signs which indicated lack of convergence.

Example 2: Old Faithful data.

For the Old Faithful data only twenty split/combine moves were accepted and the birth/death acceptance rate was 1.05%. The posterior distribution for the number of components gave a three-component mixture the largest probability. Posterior inference for other parameters was also very close to results obtained with the other split/combine

moves. Classification of the observations, using the dissimilarity matrix, gives the same allocations as the moment matching move.

Example 3: Ruspini data.

The the results for the Ruspini data showed that the sampler does not mix so well for the number of components as the with the other split/combine moves, but finds the groups efficiently, see Figure 4.8.

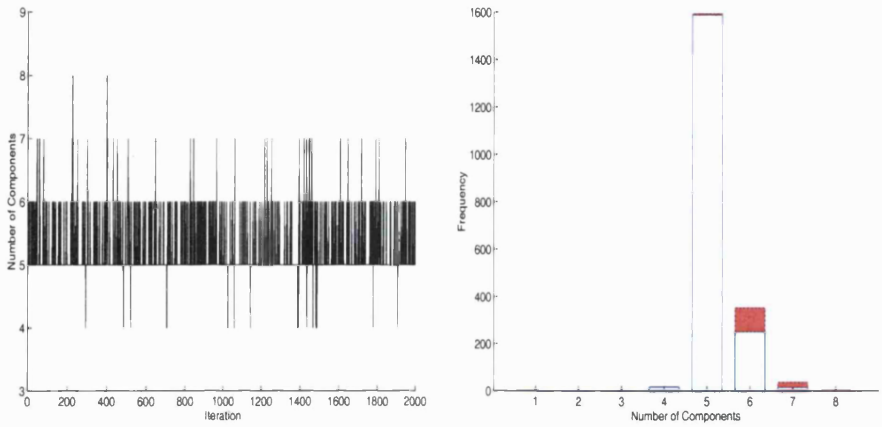


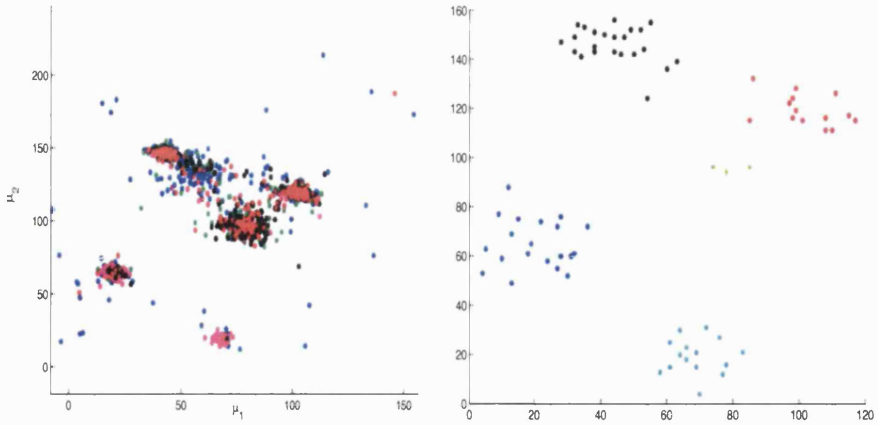
Figure 4.8: Sampled values for the number of components k by iteration and barplot of the number of components k, MST based split/combine move: Ruspini data.

This is to be expected since for this data set, the split/combine move based on PC performed well. Here the acceptance rate for the split/combine move was 0.5% and 6.5% for the birth/death move. A five-component mixture is given the largest posterior probability. The sampled values for the mean vectors and the classification of the data using the dissimilarity matrix are shown in Figure 4.9.

Example 4: Iris data.

The Iris data set is again described with only one component which was given a large variance. The sampler had a second component in 21 of the 2000 iterations, the second component was always empty. Only 6 split/combine moves were accepted and the acceptance rate for the birth/death move was around 1.3%.

To increase the chance of finding separations in a component we rescaled the data to have the same variance in all component directions. We sphered the data before obtaining the MST. That is, we considered $\mathbf{y}_i' = \mathbf{y}_i S^{-1/2}$ where $S^{1/2} = T\Lambda^{1/2}T'$ and $\Lambda = diag(\lambda_1, \ldots, \lambda_p)$ for λ_i the eigenvalues of the sample covariance matrix. Results for



(a) (b)

Figure 4.9: (a) Sampled values for the means of a 5-component mixture, MST based split/combine move: Ruspini data. (b) Classification for the Ruspini data set into five groups.

the Iris data remained unchanged. The Chebychev distance was also used to compute the length of the edges but did not improve the separation of the groups in the Iris data set.

Having observed that none of the proposed split/combine moves have allowed the reversible jump sampler to identify the three different species in the Iris data we looked carefully at the proposed split/combine moves that are accepted. We noticed that the split/combine moves based both on the principal axis and the MST did not separate efficiently the groups in the Iris data set. The moves were proposed in such a way that the observations belonging to different groups would be mixed. Therefore the proposed parameters for the new components would not help increase the likelihood.

An alternative way to divide the allocated data of the component selected to split into two groups is using the projections in the direction of the principal axis in order to search for big gaps. The projected data are ordered and the distance between adjacent projected observations is computed. We then look for the largest distance and we divide the data using this. That is we consider all the points to the left and right of the ones separated by the largest distance. Then again using equations (4.1 - 4.2) we can obtain the parameters for the new components.

With the latter move, if we begin by allocating all observations in the Iris data in one component, the moment matching type split move will separate one of the groups correctly. When it intends to separate the remaining two groups, this move is no

longer the most efficient one. We observed that once the first group was separated, the group that still had two different groups in it would be better separated considering the projections onto the principal axis to the left and right of the projected sample mean. Therefore, a mixture of different ways to separate the allocated data into two groups was implemented. The different moves were given equal probabilities and only one selected at random was carried out at each iteration. In general, different split/combine moves could be considered at each iteration of the reversible jump sampler. If one is interested in capturing any particular aspects of the analysed data, moves can be designed to accomplish this task. The acceptance of at least one of these moves shows that the algorithm was given the chance to explore different areas of the parameter space which were not so likely to be visited without the inclusion of that particular move.

However, using different split/combine moves to analyse the Iris data was not useful to separate the three groups in the Iris data set. This suggests that in terms of likelihood there is only a subtle improvement encouraging the sampler to keep only two groups.

Example 5: Lubischew's beetle data.

For the Lubischew's beetle data five split and no combine moves were accepted. The acceptance rate for the birth/death move was less that 1% this time. As shown in Figure 4.10, the sampler mixed poorly over the number of components. However, the groups were correctly identified.

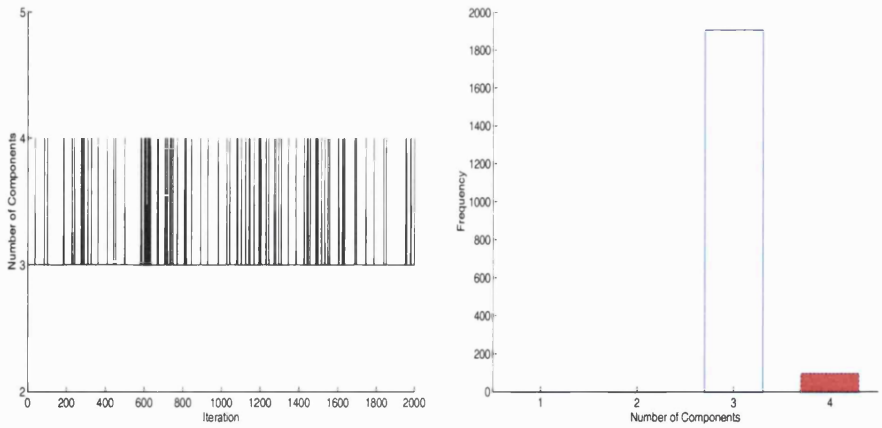
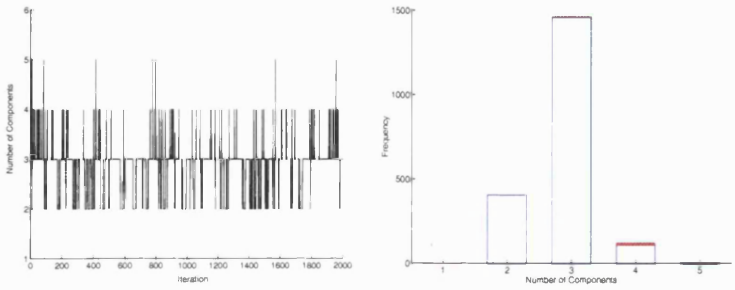


Figure 4.10: Sampled values for the number of components k by iteration, MST based split/combine move: Lubischew's beetle data.

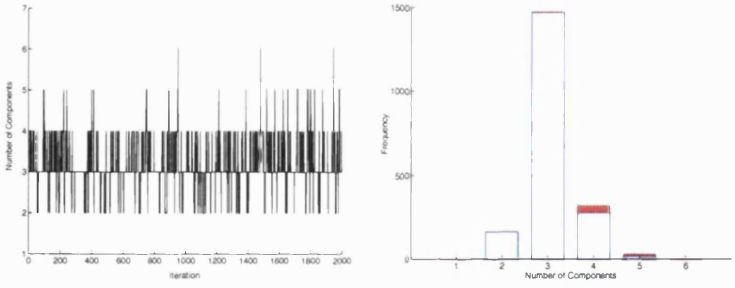
The sampled values for the mean vectors were very close to the sample values obtained for each group. The covariance matrices were also close to the sample values

Example 2: Old Faithful data.

Using the BDMCMC sampler, the data from the eruptions of the Old Faithful geyser were also described with a three-component mixture with the highest posterior probability, for both $\lambda = 1$ and $\lambda = 3$, see Figure 5.6. The mixing of the number of components is not good. Results on the number of components for this run differ slightly from the results reported in Stephens [59], where the sampler after 20000 iterations, discarding the first 10000 iterations as burn-in, has a two-component mixture with the highest posterior probability for both values of λ.



$\lambda = 1$



$\lambda = 3$

Figure 5.6: Sampled values for the number of components k by iteration and barplot of the number of components k: Old Faithful data.

In general, we observe that larger values of λ favour larger values of k, but in this example the observed data are still described adequately in terms of clustering. For both $\lambda = 1$ and $\lambda = 3$, the sampled values for the parameters are very similar and close to the value obtained from the data. The sampled values for the means of a three component mixture for $\lambda = 1$ are shown in Figure 5.7. If the data are classified using the dissimilarity matrix, it will separate the observations in exactly the same way as it was done with the output from the reversible jump sampler. The mean vectors sampled for $\lambda = 3$ give similar results and the classification remains unchanged. The covariance matrices associated to the components exhibit more variability than the rest of the

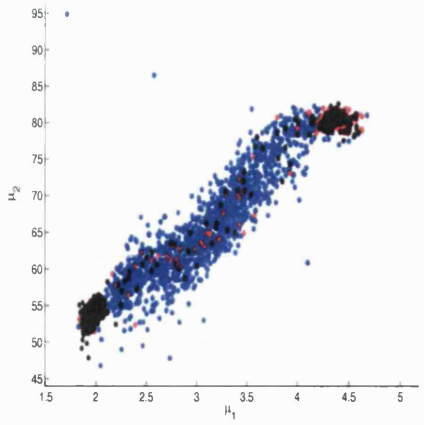
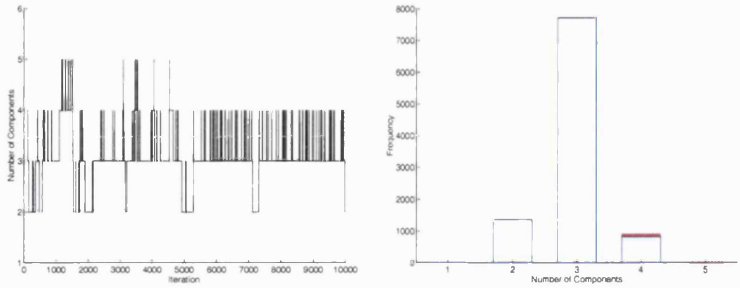


Figure 5.7: Sampled values for the mean vectors of the three-component mixture: Old Faithful data.

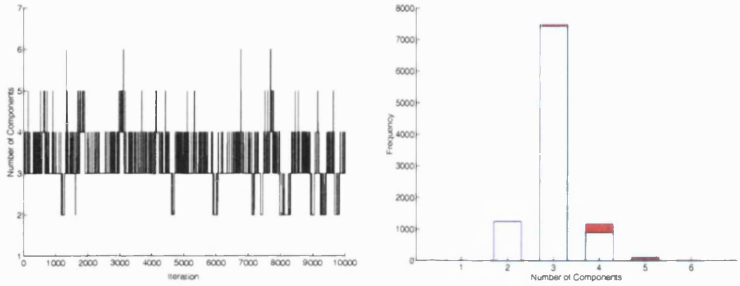
parameters, but there is no obvious tendency to use more dispersed components when λ is varied.

In order to compare the results to those reported in Stephens [59] we ran the algorithm for 20000 iterations discarding the first 10000 as burn-in for the model described by equations (5.1 and 3.12-3.14) and also for a model where another level of hierarchy is considered. That is, where ξ and κ are considered as hyperparameters with posterior full conditionals given by equations (3.27). The plots for the posterior number of components are shown in Figure 5.8. Results for this run show that a three-component mixture is fitted with the highest posterior probability for both fixed and variable κ and $\lambda = 1$ and fixed κ, $\lambda = 3$. For a variable κ and $\lambda = 3$ a four-component mixture has the highest posterior probability. We believe these are reasonable outcomes which reflect the structure of the data. The difference in results suggests that longer runs might be necessary to ensure convergence.

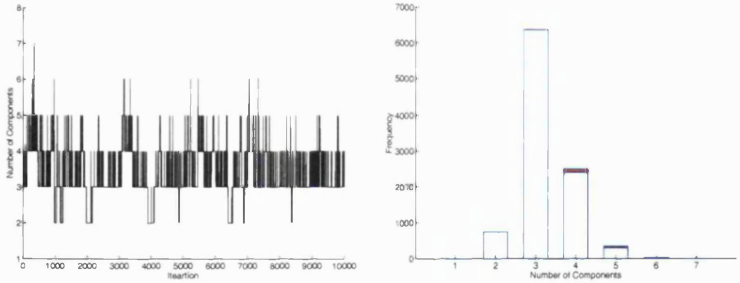
Parameter values for the mean vectors are in general very similar in the first three cases. For the last case, an additional component is added to describe the data between the two clouds of observations. Sampled values for the mean vectors and classification are shown in Figure 5.9. Covariance matrices again present more variability than the mean vectors. When $\lambda = 3$ the number of empty components is larger then when $\lambda = 1$. The model where ξ and κ are considered as hyperparameters encourages a larger number of components to be fitted.



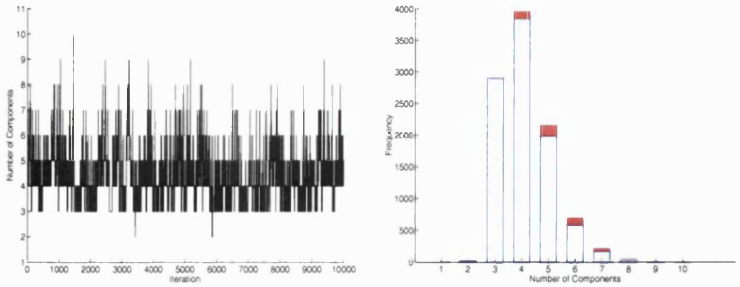
$\lambda = 1$, fixed ξ, κ



$\lambda = 3$, fixed ξ, κ

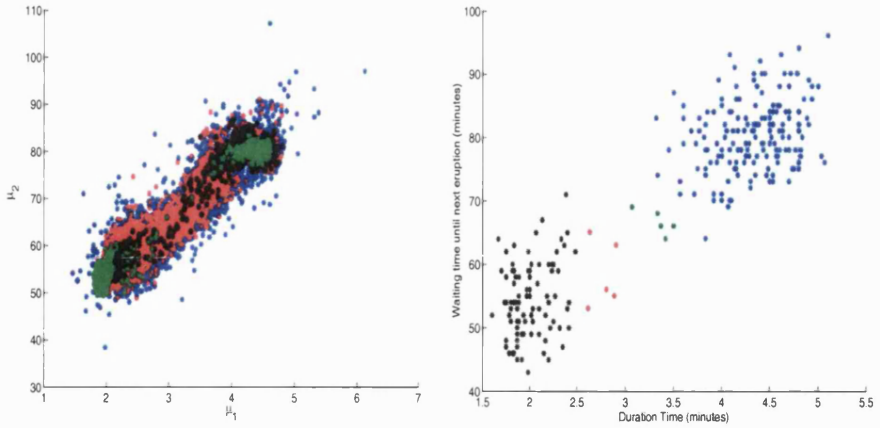


$\lambda = 1$, variable ξ, κ



$\lambda = 3$, variable ξ, κ

Figure 5.8: Sampled values for the number of components k by iteration and barplot of the number of components k: Old Faithful data, 10000 iterations after a burn-in of the same length.



(a) (b)

Figure 5.9: (a) Sampled values for the mean vectors of a four-component mixture with $\lambda = 3$ and variable ξ and κ. (b) Classification for the Old Faithful data into four groups.

Example 3: Ruspini data.

We also implemented the BDMCMC sampler for the Ruspini data set. Here the mixing over the number of components, for both $\lambda = 1$ and $\lambda = 3$, is not as good as the mixing we observed from the outputs of some of the reversible jump samplers presented in previous chapters, see Figure 5.10.

However, the groups are separated in a very similar way. The posterior probability for a five-component mixture is 0.5395 and for a four-component mixture is 0.4050, suggesting that there are five or perhaps four groups. That is, if the components in the four-component mixture model correspond approximately to those in the five-component mixture, except for a small weighted fifth component, one could consider the possibility that the small component is not indicating a different group but a departure from normality.

Once again, larger values of λ encourage more components to be fitted to the data. The resulting classification obtained from the dissimilarity matrix is given in Figure 5.11. The fifth component might be used to capture departures from normality.

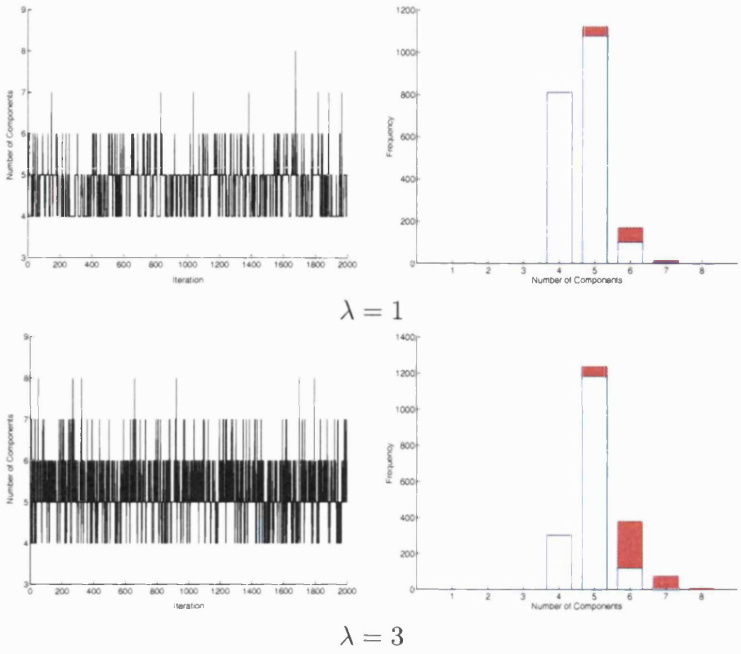


Figure 5.10: Sampled values for the number of components k by iteration and barplot of the number of components k: Ruspini data.

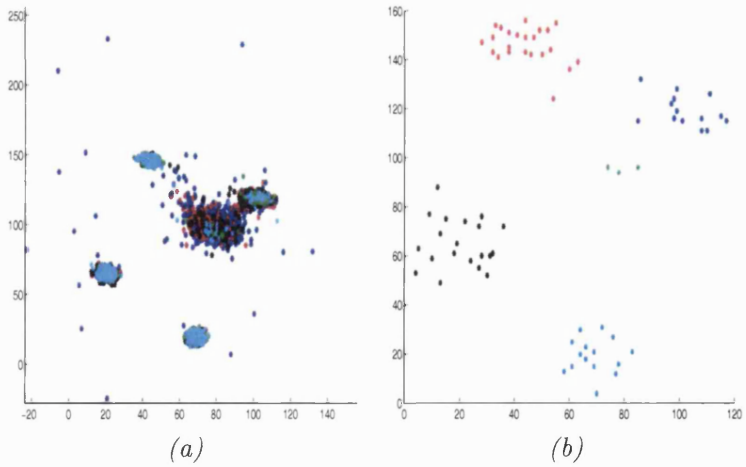


Figure 5.11: (a) Sampled values for the means of a 5-component mixture: Ruspini data. (b) Classification for the Ruspini data set into five groups.

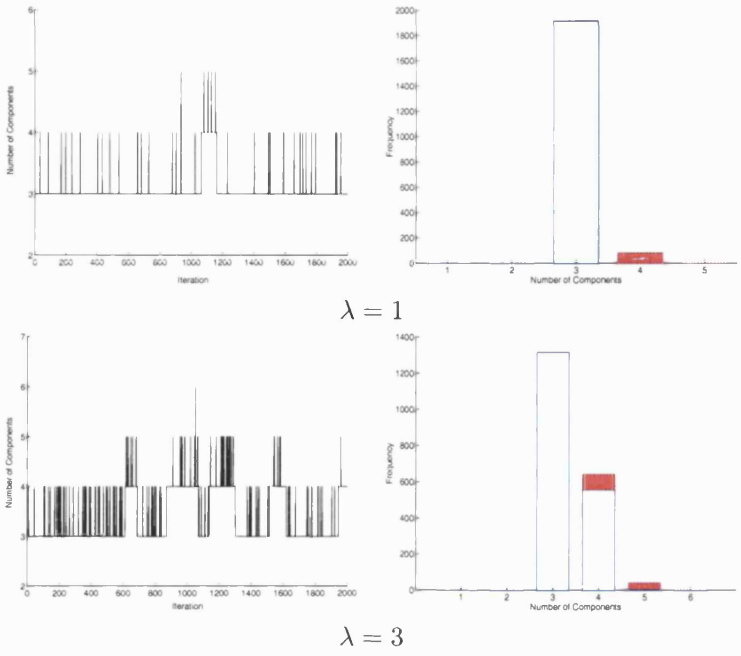


Figure 5.12: Sampled values for the number of components k by iteration and barplot of the number of components k: Iris data.

Example 4: Iris data.

The BDMCMC sampler performs a lot better for the Iris data set. With this sampler a three-component mixture has the highest posterior probability, for both $\lambda = 1$ and $\lambda = 3$, which corresponds exactly to the number of species in the data set, see Figure 5.12 . With $\lambda = 3$ larger values of k were encouraged. For $\lambda = 1$, when a 4-component mixture was fitted to the data, occasionally the fourth component was not empty. When a fifth component was present, it was always empty. For $\lambda = 3$ when a fourth component was included, in only 13% of the cases this component was empty. When five components were included in the mixture, the fifth one was empty in most cases.

The classification obtained from the dissimilarity matrix places all the observations that correspond to the *setosa* species into the first group. The second group contains all the *versicolor* plus observations $69, 71, 73, 78$ and 84 that came from the *virginica* type. The remaining 45 *virginica* iris would be the third group. It is worth mentioning that in this occasion using the *single linkage* merging criterion for the hierarchical clustering, observation 78 would be classified in the correct group. Other criteria placed the latter observation in the second group.

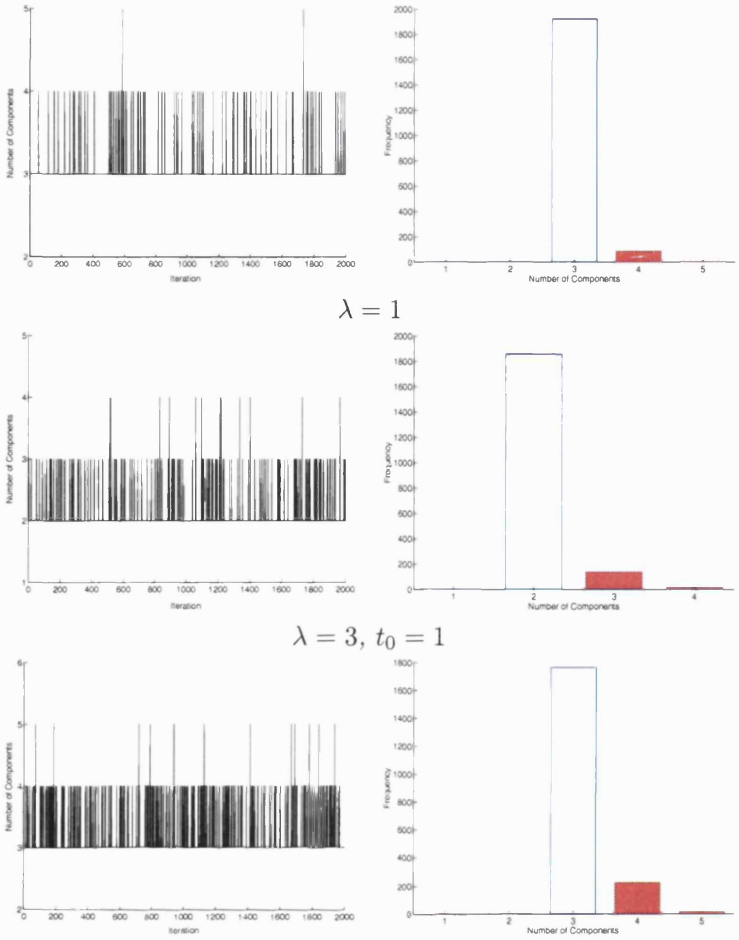


Figure 5.13: Sampled values for the number of components k by iteration and barplot of the number of components k: Lubischew's beetle data.

Example 5: Lubischew's beetle data.

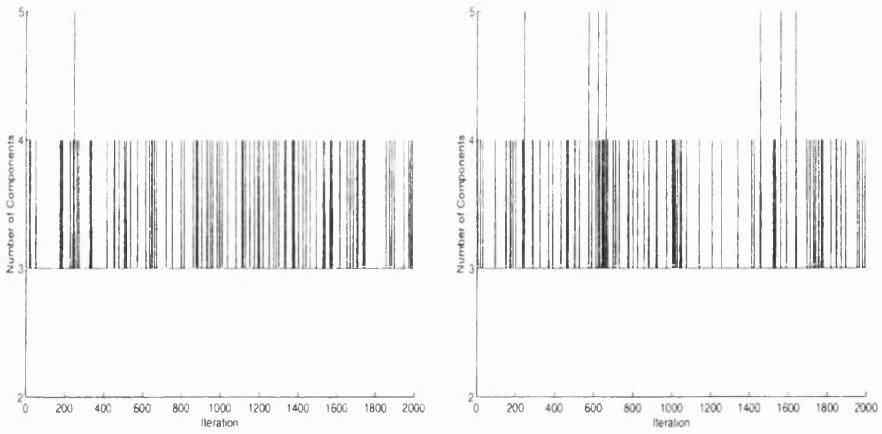
The groups in the Lubischew's beetle data were also better identified by the birth and death sampler for $\lambda = 1$. Considering all the six variables, it fitted a three-component mixture with the highest posterior probability, see Figure 5.13. Using the dissimilarity matrix, the observations were allocated into the species they belong, 21 to the *concinna*, 31 to the *heikertingeri* and 22 to the *heptapotamica*. The sampled values for the remaining parameters are close to the sample values per group observed in the data.

The fifth variable is often removed from the analysis of this data set, by doing this we obtained better results for the RJMCMC sampler in Chapter 3. For the birth and death sampler, removing the fifth variable fits a two-component mixture with the

highest posterior probability. Looking at the values sampled in both cases, we observe that the fifth variable gives the necessary information to identify the first species from the last.

However, when $\lambda = 3$ a two-component mixture was fitted with the highest posterior probability. A three-component mixture was observed in 136 iterations of which only three had a non-empty third component. In the cases where the third component was not empty, only three observations were allocated into the third component. The sampled values for the fifth element of the mean vector were very different to the corresponding values obtained from the data. We have observed that, for this example, with an initial state where $k_0 = 1$, a larger birth rate results in more events observed in the fixed time t_0 for which the birth and death is run. Deaths occur very quickly and the parameter space is not covered enough to detect the subtle difference between the first and the third beetle species. When a fourth component was included it was always empty. As the fixed time t_0 was decreased, the number of events was similar to the ones obtained for $\lambda = 1$ and $t_0 = 1$ and the resulting density fitted to the data had three components with the highest probability. When the parameter λ_b, which also corresponds to the overall birth rate, is increased, the time t_0 should be fixed to a value that allows a number of events such that data are allocated into the sampled components before they are rapidly killed. If the fixed period for the birth and death processes is reduced too much, then births do not occur at all and the parameter space is not properly explored.

In order to compare the results in this chapter with the performance of the reversible jump samplers used in Chapters 3 and 4, we ran the BDMCMC sampler using a uniform prior distribution in $\{1, \ldots, k_{max} = 30\}$ for the number of components k. The algorithm was run for a burn-in period of 200000 iterations followed by 100000 monitored iterations thinned every 50.



(a) (b)

Figure 5.14: Simulated data. *(a)* Sampled values for the number of components by iteration ($\lambda_b = 1$). *(b)* Sampled values for the number of components by iteration ($\lambda_b = 3$). *(b)*

In general, the results obtained using a uniform prior were very similar to those obtained using a truncated Poisson prior for k, see Figures 5.14- 5.18. The conclusions in terms of classification using a dissimilarity matrix and a hierarchical clustering remained unchanged.

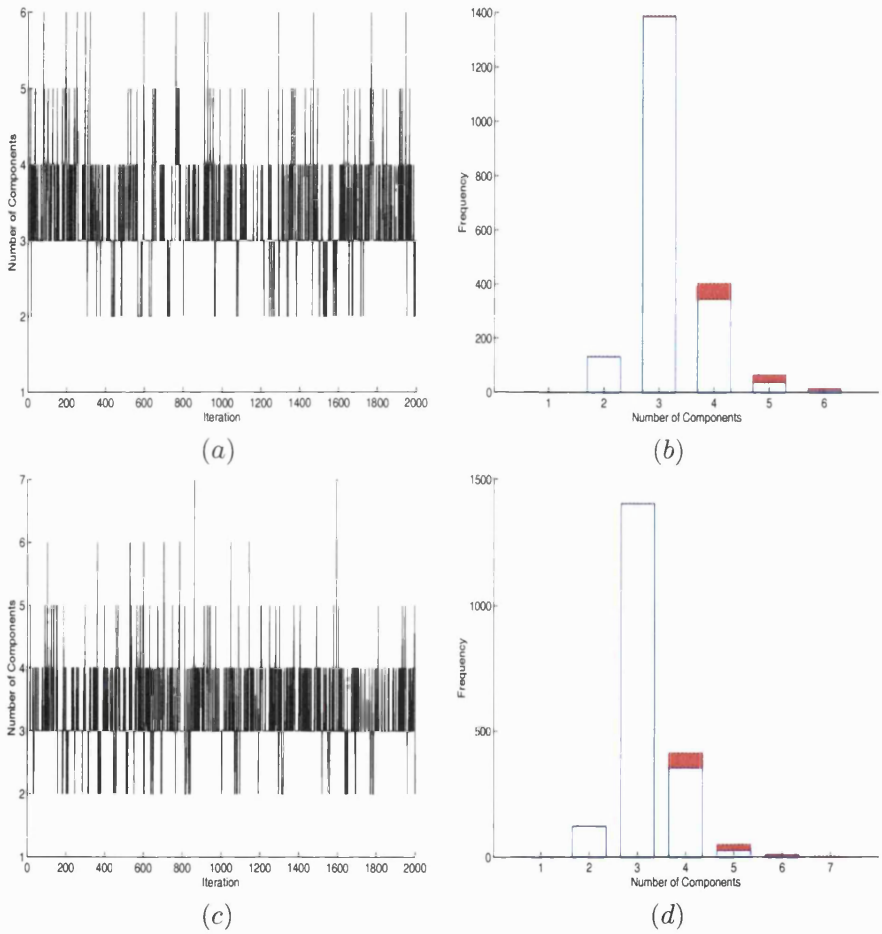


Figure 5.15: Old Faithful data. *(a)* Sampled values for the number of components by iteration ($\lambda_b = 1$). *(b)* Barplot for the number of components ($\lambda_b = 1$). *(c)* Sampled values for the number of components by iteration ($\lambda_b = 3$). *(d)* Barplot for the number of components ($\lambda_b = 3$).

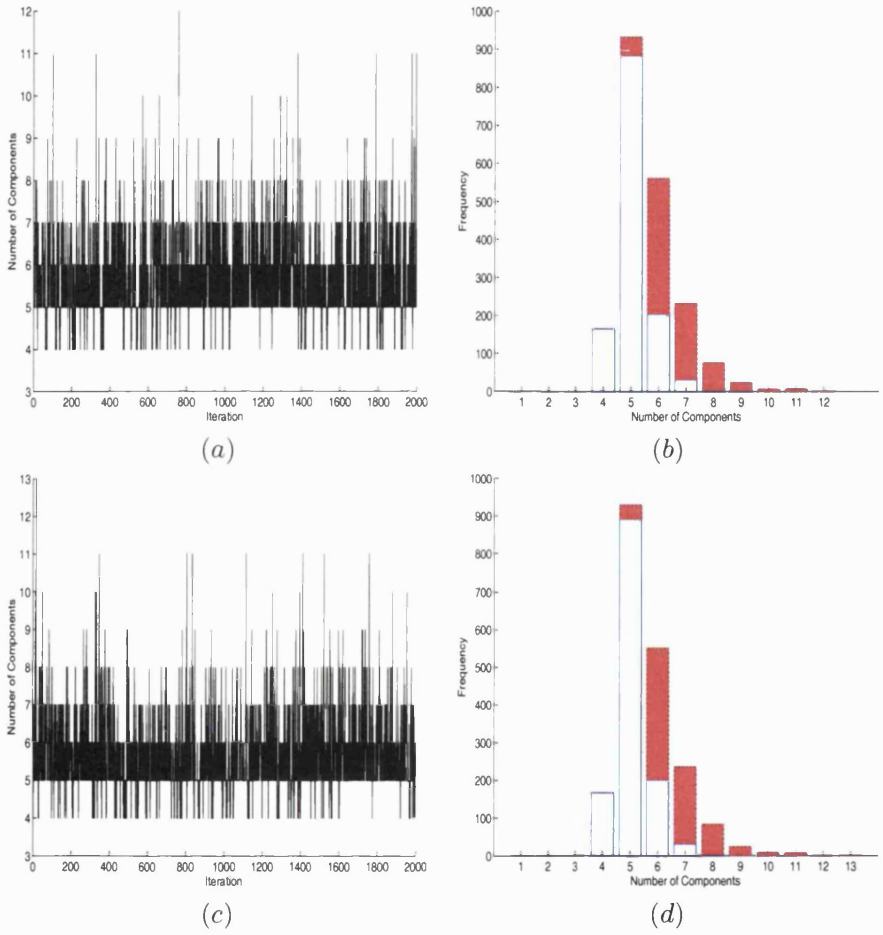


Figure 5.16: Ruspini data. *(a)* Sampled values for the number of components by iteration ($\lambda_b = 1$). *(b)* Barplot for the number of components ($\lambda_b = 1$). *(c)* Sampled values for the number of components by iteration ($\lambda_b = 3$). *(d)* Barplot for the number of components ($\lambda_b = 3$).

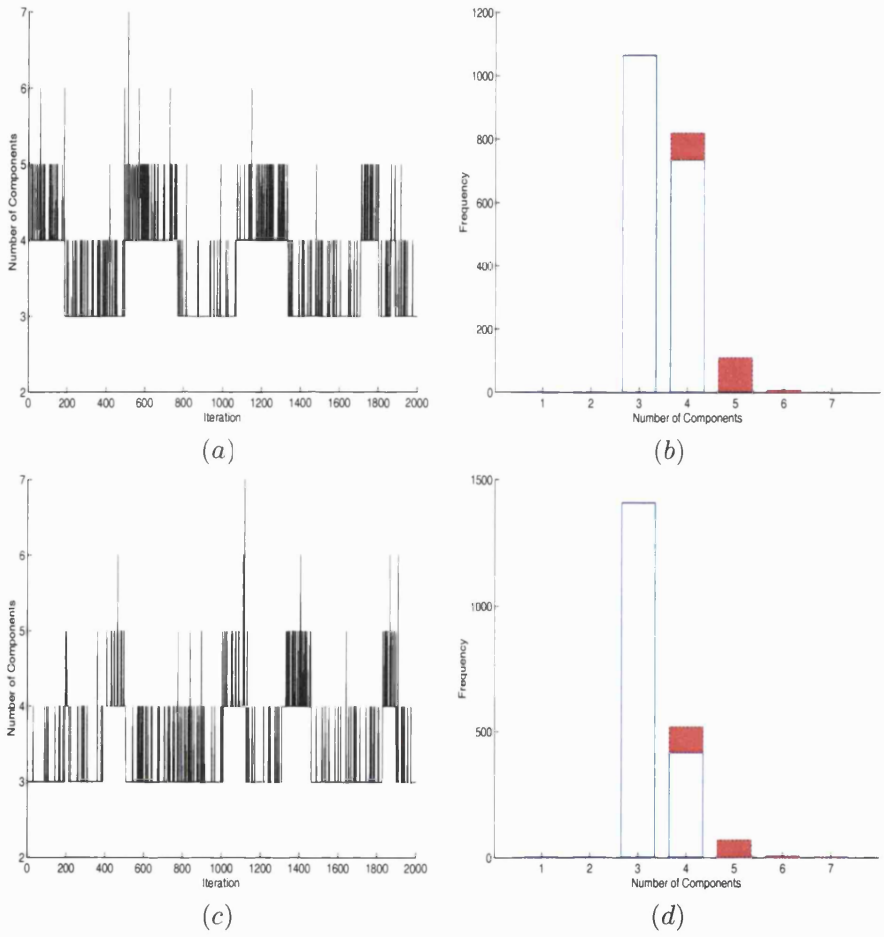


Figure 5.17: Iris data. *(a)* Sampled values for the number of components by iteration ($\lambda_b = 1$). *(b)* Barplot for the number of components ($\lambda_b = 1$). *(c)* Sampled values for the number of components by iteration ($\lambda_b = 3$). *(d)* Barplot for the number of components ($\lambda_b = 3$).

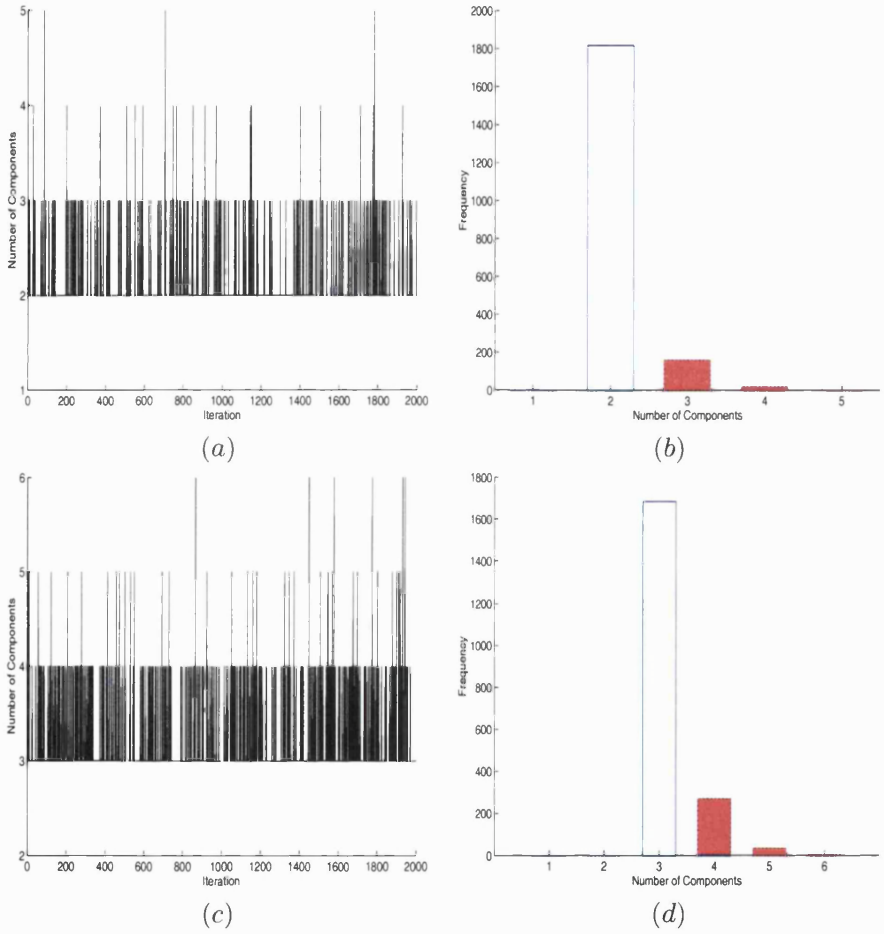


Figure 5.18: Beetle data (six variables). *(a)* Sampled values for the number of components by iteration ($\lambda_b = 1$). *(b)* Barplot for the number of components ($\lambda_b = 1$). *(c)* Sampled values for the number of components by iteration ($\lambda_b = 3$). *(d)* Barplot for the number of components ($\lambda_b = 3$).

Straightforward BDMCMC has proved less problematic in practice than RJMCMC, even for the higher dimensional cases. Results for the examples we have considered suggest that the BDMCMC sampler explores more unlikely areas of the parameter space and this helps describe the groups in a data set. The latter could be seen for example in the Iris data set, where a three component mixture is given high posterior probability. The posterior number of components fitted to describe a data set increases when there is a departure from normality of one or many of the groups.

5.3 Data driven prior

Recall that the prior distribution given to the mean vectors of the components in the mixture model is a multivariate normal distribution $N_p(\xi, \kappa^{-1})$. Richardson and Green [51] and Stephens [59] discussed the effect of this prior on posterior inference. They indicate that the value chosen for κ has a subtle effect on the posterior distribution of the number of components, k. While reducing the value of κ encourages more components, a further reduction will result in less fitted components. That is, very large and very small values lead to informative priors on k.

The prior given to the mean vectors also determine the areas of the parameter space explored by the BDMCMC sampler. To pursue efficiency the sampler needs to move towards areas where components are more likely to survive and we would then expect to obtain less empty components. We are interested in using the data $\{\mathbf{y}_j\}$ to concentrate the prior distribution on areas that have higher probability of getting data allocated into them, that is we want to have a prior distribution for the component mean vectors that would allow the sampler to move quickly to areas of high probability. Although in principle the information provided by the data is being used to define the prior on the means and to update this prior, we are only looking for a gain in efficiency.

Consider a prior for the mean vector of the multivariate normal distributions as a mixture of multivariate normal distributions given by

$$p(\boldsymbol{\mu}) = \sum_{j=1}^{n} \frac{1}{n} N_p(\mathbf{y}_j, rI),$$

where the value of r must be such that it is possible to compute the resulting posterior full conditional. The posterior full conditional distribution is a mixture of multivariate normal distributions. As the value of r decreases, the exponent in the expression for

the posterior mixing proportions increases. This leads to numerical problems when trying to compute the needed constant to transform the mixing proportions so that they add up to one. The posterior full conditional is given by

$$p(\boldsymbol{\mu}|\cdots) = \sum_{j=1}^{n} s_j N_p\left((rI)^{-1} + n_i\Sigma_i^{-1})^{-1}((rI)^{-1}\mathbf{y}_j + n_i\Sigma^{-1}\overline{\mathbf{y}}_i), ((rI)^{-1} + n_i\Sigma_i^{-1})^{-1}\right),$$

where $n_i = \#\{j : z_j = i\}$ and $\overline{\mathbf{y}}_i = 1/n_i \sum_{z_j=i} \mathbf{y}_j$. The mixing proportions s_j are such that

$$s_j \propto \exp\left\{-\frac{1}{2}\left[\mathbf{y}_j^T(rI)^{-1}\mathbf{y}_j - \frac{1}{r^2}\mathbf{y}_j((rI)^{-1} + n_i\Sigma_i^{-1})^{-1})\mathbf{y}_j^T - \frac{2n_i}{r}\Sigma_i^{-1}\overline{\mathbf{y}}_i((rI)^{-1} + n_i\Sigma_i^{-1})^{-1}\mathbf{y}_j\right]\right\}$$

and $\sum_{j=1}^{n} s_j = 1$.

5.3.1 Examples

We describe the results for two of the examples we have followed: the Old Faithful data set and the Ruspini data set. Results for the rest of the data sets were similar to results of these examples.

In each case the BDMCMC sampler with the data informed prior for the mean vectors was run for 300000 iterations, discarding the first 200000 iterations as a burn-in period and thinning the remaining 100000 every 50 iterations. Values given to r correspond to $\xi_{min}/2$, where

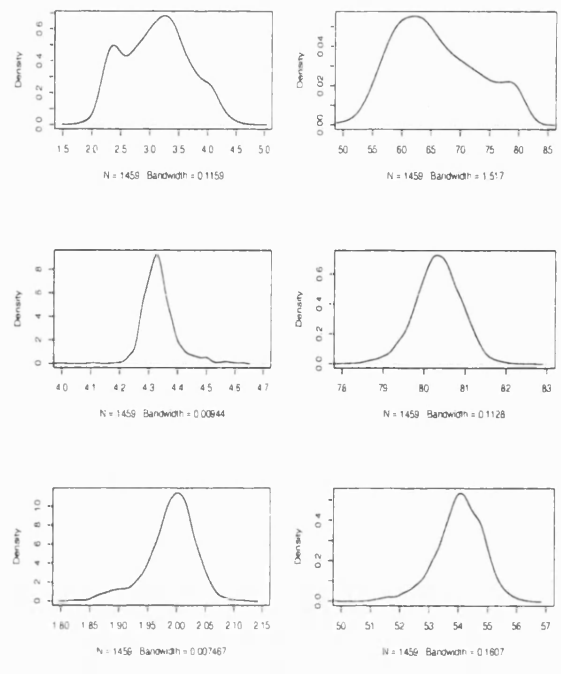
$$\xi_{min} = \min_j\{\xi_1, \ldots, \xi_p\},$$

and ξ_j denotes the range of the observed variables for the j-th feature.

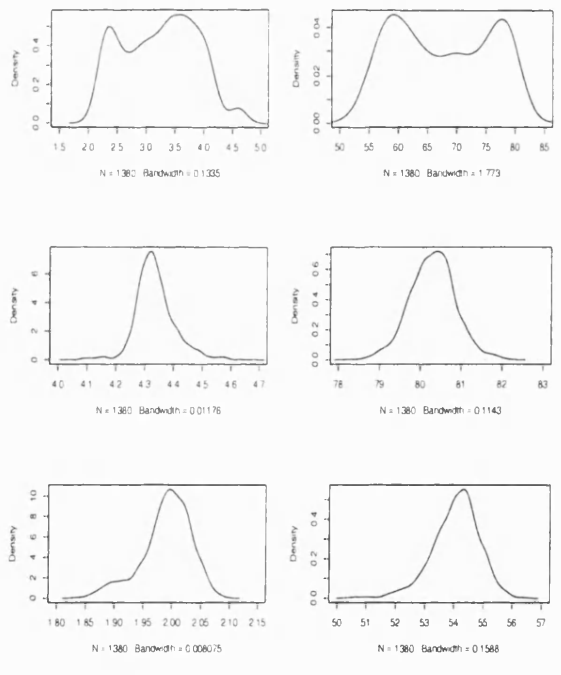
Old Faithful data

In this case, the number of empty components is slightly smaller and the changes in the posterior distribution for k are negligible. The classification obtained using the dissimilarity matrices did not change when using a data driven prior for the mean vectors.

The posterior densities of the individual entries of the mean vectors for the BDMCMC samplers with a multivariate normal prior density and a data driven prior density are shown in Figure 5.19. We observe that the densities for the first component show



(*a*)



(*b*)

Figure 5.19: Posterior density for the elements of the mean vectors (columns) in a three-component mixture (rows) : (a) BDMCMC with a multivariate normal prior for the mean vector. (b) BDMCMC with a data informed prior for the mean vectors. Old Faithful data.

two modes, these modes are clearly shown with the data driven normal prior compared to the straightforward normal prior. For the rest of the components the densities of the corresponding means do not show important differences.

Ruspini data

The BDMCMC sampler with a data driven prior for the mean vectors showed a considerable reduction on the number of empty components for the Ruspini data set, approximately 3% less than with the straightforward normal prior. A five-component mixture is fitted with highest posterior probability in both cases and the classification based on the dissimilarity matrix did not change with the use of a data driven prior for the mean vectors.

The posterior densities for the elements of the mean vectors of the five-component mixture are displayed in Figure 5.20 for both samplers. In general, the densities for the individual parameters show a very similar behaviour. For some entries of the mean vectors, the sampled distributions have a slightly heavier tail for the BDMCMC sampler with a normal prior compared to the BDMCMC sampler with a data driven prior.

Having analysed the effect of the data driven prior for the mean vectors of the normal components, we believe there is no significant gain in efficiency and to preserve the simplicity of a conjugate posterior we will assign a normal prior for the mean vectors in the work that we present in Chapter 7.

Thus far, we have observed some differences in terms of the posterior distribution of the number of components k when using different trans-dimensional samplers. The groups in some of the examples we have followed have been identified consistently by all the samplers. However, we have not seen this for all cases. The Iris data set is difficult to describe by a multivariate normal mixture model, the BDMCMC was the only sampler giving high posterior probability to a three-component mixture. We believe the main problem is that in practical clustering there is a real difficulty in associating each cluster with a single component in the mixture model. We will approach this by associating a group to a submixture of components in Chapter 7.

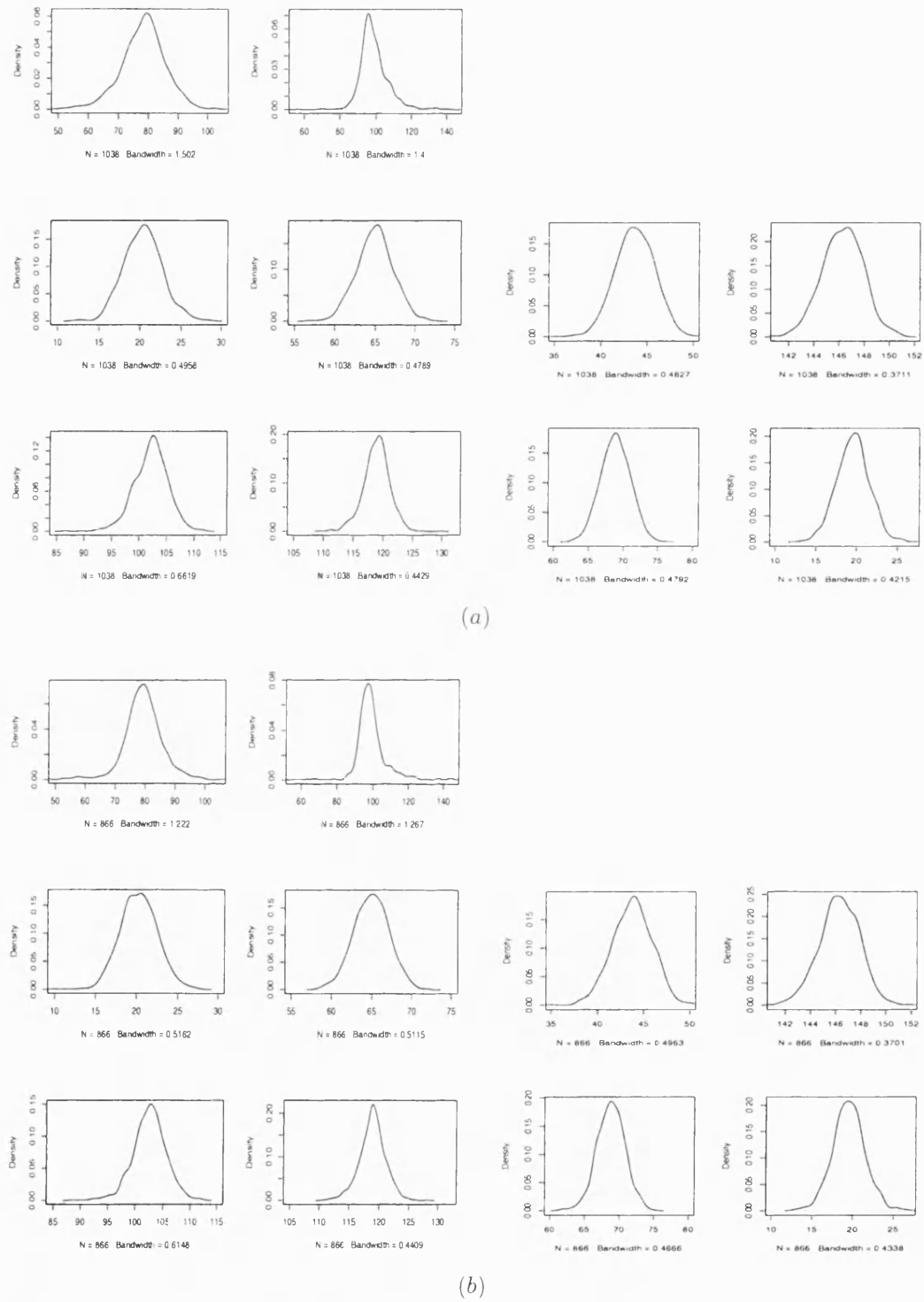


Figure 5.20: Posterior density for the elements of the mean vectors (columns) in a five-component mixture (rows) : (a) BDMCMC with a multivariate normal prior for the mean vector. (b) BDMCMC with a data informed prior for the mean vectors. Ruspini data.

CHAPTER 6

Convergence assessment for trans-dimensional Markov chain Monte Carlo samplers

The present chapter is dedicated to the discussion the convergence of the Markov chain Monte Carlo samplers where the parameter space can change dimension. In previous chapters, we have used RJMCMC and BDMCMC methodologies to obtain samples from the posterior distribution of the parameters of interest in the mixture model and used those samples for inference. We have considered reasonably long runs and followed Richardson and Green[51] and Stephens[59] in concentrating on the mixing over the number of components. Then, conditional on the number of components, we looked at the behaviour of the remaining parameters individually. In this chapter, we will follow recent work by Castelloe and Zimmerman [13] to assess convergence for the RJMCMC and the BDMCMC samplers.

As authors often state, we are interested in finding evidence to support the hypothesis that the chain we are using to make inference has reached equilibrium. In other words, we want to know if the samples are being generated from the correct distribution and if the parameter space has been adequately covered.

The assessment of convergence in trans-dimensional MCMC samplers is particularly difficult. As Brooks *et al* [10] and Castelloe and Zimmerman [13] highlight, the challenge lies in finding parameters that retain the same interpretation throughout different models. In the following sections, we briefly present the nonparametric convergence as-

sessment discussed in Brooks *et al* [10], pointing out the difficulties encountered in high dimensional problems. Then we will follow the analysis of variance type convergence assessment given by Castelloe and Zimmerman [13] to assess convergence of the RJMCMC and the BDMCMC samplers and report the results for some of the examples we have followed throughout this work.

Generally speaking, we will select a group of observations from the data set and monitor the parameters of the components to which these data are allocated at each iteration and for different chains. Notice that this set of parameters retains a coherent interpretation across models, a crucial feature for the convergence assessment of trans-dimensional samplers. The selected observations are chosen so that their behaviour is expected to vary across sweeps of the sampler in different ways. Namely, we will look for data that are between two clusters, that is between potential competitors when allocating the observations, potential outliers and also data near the centre of a cluster. The convergence assessment looks for evidence that indicates lack of convergence for the set of monitored parameters both across iterations and across chains.

6.1 Nonparametric convergence assessment

In the context of both clustering and mixture modelling, Brooks *et al* [10] considered the problem of allocating each observation $y_i, \ldots, y_n$ into one of an unknown number of components $s < s_{max}$. Once $s_{max} \leq n$ is fixed, a model is described by an ordered vector $M_i = [m_i(1), \ldots, m_i(n)] \in \mathcal{M}$, where $\mathcal{M}$ is the set of possible models. Here, $m_i(j) = l$ if observation y_j is assigned to component l under model M_i, we refer to M_i as the allocation vector. For simplicity of notation, the authors assume components are ordered from 0 to $s_{max} - 1$. The main objective is then to estimate the probability of different models.

Using the output of a trans-dimensional MCMC sampler it is possible to estimate the probability of different models. The natural estimator for the probability $Pr(M = M_i) = P_i$ from the T iterations is given by

$$\hat{P}_{i,T} = \frac{1}{T} \sum_{t=1}^{T} \mathbf{I}_{\{M_i\}}, \quad \text{for all} \quad M_i \in \mathcal{M}.$$

The assessment of convergence is based on J independent Markov chains each one

of length T. For each chain $j = 1, \ldots, J$ the probability mass function for models is estimated. The estimate is denoted as $\hat{\mathbf{P}}_T^j = (\hat{P}_{1,T}^j, \hat{P}_{2,T}^j, \ldots, \hat{P}_{c,T}^j)$. Since all chains come from the same stationary distribution, the estimates of the probability mass functions from the different chains should be similar, for sufficiently large T and if the chains can be assumed to have converged. The output of the sampler is not independent and the thinning of the chain is advised to reduce dependence.

However, in Brooks *et al* [10] the identifiability problems encountered in mixture modelling are not considered. In high dimensions, we have not placed any constraints on the parameters and the problem of label switching will need to be addressed in such a way that the computation of the proposed model identifier would be meaningful. That is, we need to address other issues to be able to assess the convergence of the samplers. The allocation vector could for example, be renamed at each iteration to identify each model uniquely but this would again require the relabelling of all the associated parameters at each iteration, a computationally demanding task.

6.2 ANOVA type convergence assessment

An alternative approach is described in Castelloe and Zimmerman [13], as an extension of Gelman and Rubin's [30] technique, based on an analysis of variance (ANOVA) type approach. Gelman and Rubin's method requires several chains to be run, " chain" is considered a factor, and the ratio of a pooled variance estimate and a within-chain variance estimate is computed. When the two variances are comparable, one can consider the chains as realisations of a common distribution, presumably the correct limiting distribution. The method depends on the absence of other significant factors, Castelloe and Zimmerman [13] suggest that in the trans-dimensional MCMC sampler, an indicator of the parameter space, hereon referred to as " model ", could be considered as a factor.

Consider the output of the sampler as a parameter vector $\boldsymbol{\theta} \in \Theta$, with some $\theta' \in \boldsymbol{\theta}$ indexing the " model ". Let $\boldsymbol{\theta}_* \in \boldsymbol{\theta}$ be a subset of $\boldsymbol{\theta}$ which retains the same interpretation across models ($\theta' \notin \boldsymbol{\theta}_*$). In the mixture model context we will consider $\theta' = k$, the number of components of the mixture model in the current iteration. Suppose $C > 1$ chains of a trans-dimensional MCMC sampler, which are started from overdispersed states, are run for the same number of sweeps. Then, a number m of successive over-

lapping *batches* for each chain are analysed. The length of these batches increases and each length is a multiple of a base batch length b.

For simplicity of notation only one batch of size qb for some q is considered, so we have the parameter vectors

$$\left(\boldsymbol{\theta}_{*1}^{(qb+1)},\ldots,\boldsymbol{\theta}_{*1}^{(2qb)}\right),\ldots,\left(\boldsymbol{\theta}_{*C}^{(qb+1)},\ldots,\boldsymbol{\theta}_{*C}^{(2qb)}\right).$$

Let T denote the batch size and M the total number of different models visited by any chain for this batch. We will use the following notation:

$$\begin{aligned}
\boldsymbol{\theta}_* &= \text{vector of parameters retaining} \\
&\quad \text{the same interpretation across models,} \\
\theta &= \text{arbitrary scalar } \theta \in \boldsymbol{\theta}_*, \\
C &= \text{the number of chains,} \\
T &= \text{the batch size,} \\
M &= \text{number of distinct models visited by any chain,} \\
\boldsymbol{\theta}_{*cm}^{r} &= \text{value of } \boldsymbol{\theta}_* \text{ for the } r^{th} \text{ occurrence} \\
&\quad \text{of model } m \text{ in chain } c, \\
R_{cm} &= \text{number of times model } m \text{ occurred in chain } c, \\
R_{\cdot m} &= \sum_{c=1}^{C} R_{cm}, \\
\overline{\boldsymbol{\theta}}_{*cm}^{\cdot} &= \frac{1}{R_{cm}} \sum_{r=1}^{R_{cm}} \boldsymbol{\theta}_{*cm}^{r}, \\
\overline{\boldsymbol{\theta}}_{*c\cdot}^{\cdot} &= \frac{1}{T} \sum_{m=1}^{M} \sum_{r=1}^{R_{cm}} \boldsymbol{\theta}_{*cm}^{r}, \\
\overline{\boldsymbol{\theta}}_{*\cdot m}^{\cdot} &= \frac{1}{R_{\cdot m}} \sum_{c=1}^{C} \sum_{r=1}^{R_{cm}} \boldsymbol{\theta}_{*cm}^{r}, \\
\overline{\boldsymbol{\theta}}_{*\cdot\cdot}^{\cdot} &= \frac{1}{CT} \sum_{c=1}^{C} \sum_{m=1}^{M} \sum_{r=1}^{R_{cm}} \boldsymbol{\theta}_{*cm}^{r}.
\end{aligned}$$

The convergence diagnostic aims to find conditions that would indicate that convergence has not been reached. Some aspects it will detect are:

- variation between chains;

- an interaction between models and chains, which indicates between-model variation that differs from chain to chain;
- significant differences in frequencies of model visits from one chain to another.

Castelloe and Zimmerman [13] based their convergence diagnostics on the following quantities, which could be interpreted as: $\widehat{V}$, the total variation; Wc, variation within chains; Wm variation within models and $WmWc$, variation within models and chains. The corresponding expressions are given by

$$\widehat{V}(\theta) = \frac{1}{CT-1}\sum_{c=1}^{C}\sum_{m=1}^{M}\sum_{r=1}^{R_{cm}}(\theta_{cm}^{r}-\overline{\theta}_{..})^2, \tag{6.1}$$

$$Wc(\theta) = \frac{1}{C(T-1)}\sum_{c=1}^{C}\sum_{m=1}^{M}\sum_{r=1}^{R_{cm}}(\theta_{cm}^{r}-\overline{\theta}_{c.})^2, \tag{6.2}$$

$$Wm(\theta) = \frac{1}{CT-M}\sum_{c=1}^{C}\sum_{m=1}^{M}\sum_{r=1}^{R_{cm}}(\theta_{cm}^{r}-\overline{\theta}_{.m})^2, \tag{6.3}$$

$$WmWc(\theta) = \frac{1}{C(T-M)}\sum_{c=1}^{C}\sum_{m=1}^{M}\sum_{r=1}^{R_{cm}}(\theta_{cm}^{r}-\overline{\theta}_{cm})^2. \tag{6.4}$$

Castelloe and Zimmerman [13] considered the output from a trans-dimensional MCMC sampler as a collection of observations from a factorial design[1], where the factors are " chain " and/or " model ". Then, an ANOVA is used to assess the significance of factors and interactions. Although the assumptions of independence and normality in general do not hold, Castelloe and Zimmerman [13] pointed out that the effects of dependence are likely to be at least approximately cancelled out since they focus on ratios of mean squares, an implicit assumption that has been made for other convergence diagnostics in literature, for example in Gelman and Rubin [30], Brooks and Gelman [8] and Brooks and Giudici [9].

The " model " was represented as the fixed factor and " chain " as a random factor. The authors indicate that the conclusions reached when considering " model " as a random factor and " chain " as a fixed factor differ only in the description of the effects and some minor coefficients. The following analyses of variance were carried out

[1] In a factorial design the effects of a number of different treatments are investigated simultaneously. The treatments consist of all combinations that can be formed from different factors. See Cochran and Cox [17] for a detailed presentation.

1. One-way ANOVA with factor chain (random) balanced[2]

$$\theta^r_{*cm} = \mu + \alpha_c + e^r_{cm(1)}, \tag{6.5}$$

2. One-way ANOVA with factor model (fixed) unbalanced

$$\theta^r_{*cm} = \mu + \beta_m + e^r_{cm(2)}, \tag{6.6}$$

3. Two-way ANOVA with factors model (fixed), chain (random) and chain-model interaction (random,unrestricted) balanced across chain only

$$\theta^r_{*cm} = \mu + \alpha_c + \beta_m + (\alpha\beta)_{cm} + e^r_{cm(3)}, \tag{6.7}$$

where

$$\begin{aligned}
\alpha_c &\overset{i.i.d.}{\sim} N(0, \sigma^2_{ch}), \\
e^r_{cm(1)} &\overset{i.i.d.}{\sim} N(0, \sigma^2_{er(ch)}), \\
\sum_{m=1}^{M} \beta_m &= 0, \\
e^r_{cm(2)} &\overset{i.i.d.}{\sim} N(0, \sigma^2_{er(mo)}), \\
\sigma^2_{mo} &= \frac{1}{M-1} \sum_{m=1}^{M} \beta^2_m, \\
(\alpha\beta)_{cm} &\overset{i.i.d.}{\sim} N(0, \sigma^2_{ch\times mo}), \\
e^r_{cm(3)} &\overset{i.i.d.}{\sim} N(0, \sigma^2_{er(ch\times mo)}).
\end{aligned}$$

Notice that the three error terms are labelled differently because they are not equiv-

[2]A balanced design is one where any treatment is preceded by each of the other treatments.

alent. From the ANOVA's it may be established that

$$\begin{aligned} \widehat{V} &\equiv \text{MS}_{\,tot} \text{ for ANOVA 1,} \\ Wm &\equiv \text{MS}_{er(ch)} \text{ for ANOVA 1,} \\ Wc &\equiv \text{MS}_{er(mo)} \text{ for ANOVA 2,} \\ WmWc &\equiv \text{MS}_{er(ch\times mo)} \text{ for ANOVA 3.} \end{aligned}$$

After deriving the expected mean-squares for the three ANOVA models, see Castelloe and Zimmeran [13] for an exhaustive exposition, it can be shown that the ratio $\frac{\mathbf{E}\widehat{V}}{\mathbf{E}Wc} \geq 1$, with $\frac{\mathbf{E}\widehat{V}}{\mathbf{E}Wc} = 1$ indicating the absence of a chain effect. The greater the value of this ratio, the stronger the chain effect. Both, numerator and denominator, stabilise as $T \rightarrow \infty$. It can also be shown that the ratio $\frac{\mathbf{E}Wm}{\mathbf{E}WmWc} \geq 1$, with $\frac{\mathbf{E}Wm}{\mathbf{E}WmWc} = 1$ indicating: (a) the absence of chain effect, (b) the absence of chain×model interaction and (c) either no model effect or equality of the set of within-chain model frequencies across chains or both. The greater the violation of any of these aspects, the larger this ratio becomes. Here the authors emphasised that the sensitivity of this ratio to the violation of the mentioned aspects is not yet fully understood in terms of the relative weight of the three aspects as $T \rightarrow \infty$.

Therefore, the convergence diagnostics proposed which are based on both ratios, are called *potential scale reduction factors (PSRF)*. Using these ratios any potential violations of convergence is monitored. For a parameter vector $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)^T$ using expressions (6.1 - 6.4), we have

$$PSRF1(\theta_i) = \frac{\mathbf{E}\widehat{V}(\theta_i)}{\mathbf{E}Wc(\theta_i)}, \tag{6.8}$$

$$PSRF2(\theta_i) = \frac{\mathbf{E}Wm(\theta_i)}{\mathbf{E}WmWc(\theta_i)}. \tag{6.9}$$

A multivariate version is also defined to monitor the entire vector rather than considering each element separately. The corresponding multivariate versions for expressions

(6.1 - 6.4) are given as

$$\widehat{V}(\boldsymbol{\theta}_*) = \frac{1}{CT-1}\sum_{c=1}^{C}\sum_{m=1}^{M}\sum_{r=1}^{R_{cm}}(\boldsymbol{\theta}^r_{*cm} - \overline{\boldsymbol{\theta}}_{*\cdot\cdot})(\boldsymbol{\theta}^r_{*cm} - \overline{\boldsymbol{\theta}}_{*\cdot\cdot})^T, \tag{6.10}$$

$$Wc(\boldsymbol{\theta}_*) = \frac{1}{C(T-1)}\sum_{c=1}^{C}\sum_{m=1}^{M}\sum_{r=1}^{R_{cm}}(\boldsymbol{\theta}^r_{*cm} - \overline{\boldsymbol{\theta}}_{*c\cdot})(\boldsymbol{\theta}^r_{*cm} - \overline{\boldsymbol{\theta}}_{*c\cdot})^T, \tag{6.11}$$

$$Wm(\boldsymbol{\theta}_*) = \frac{1}{CT-M}\sum_{c=1}^{C}\sum_{m=1}^{M}\sum_{r=1}^{R_{cm}}(\boldsymbol{\theta}^r_{*cm} - \overline{\boldsymbol{\theta}}_{*\cdot m})(\boldsymbol{\theta}^r_{*cm} - \overline{\boldsymbol{\theta}}_{*\cdot m})^T, \tag{6.12}$$

$$WmWc(\boldsymbol{\theta}_*) = \frac{1}{C(T-M)}\sum_{c=1}^{C}\sum_{m=1}^{M}\sum_{r=1}^{R_{cm}}(\boldsymbol{\theta}^r_{*cm} - \overline{\boldsymbol{\theta}}_{*cm})(\boldsymbol{\theta}^r_{*cm} - \overline{\boldsymbol{\theta}}_{*cm})^T. \tag{6.13}$$

The corresponding multivariate scale reduction factors are

$$MPSRF1(\boldsymbol{\theta}_*) = \text{maximum eigenvalue of } [Wc(\boldsymbol{\theta}_*)]^{-1}\widehat{V}(\boldsymbol{\theta}_*), \tag{6.14}$$

$$MPSRF2(\boldsymbol{\theta}_*) = \text{maximum eigenvalue of } [WmWc(\boldsymbol{\theta}_*)]^{-1}Wm(\boldsymbol{\theta}_*). \tag{6.15}$$

In Castelloe and Zimmeran [13], it was shown that

$$MPSRF1(\boldsymbol{\theta}_*) \geq \max_i PSRF1(\theta_i) \quad \text{and}$$
$$MPSRF2(\boldsymbol{\theta}_*) \geq \max_i PSRF2(\theta_i).$$

6.2.1 Convergence assessment

To sum up, the implementation of the convergence assessment is given by the following steps:

1. Identify a parameter $\theta' \in \boldsymbol{\theta}$ which is the " model " indicator and select a subset parameter vector $\boldsymbol{\theta}_* = (\theta_1, \ldots, \theta_p)^T \in \boldsymbol{\theta}$ which retains the same interpretation across θ' and $\theta' \notin \boldsymbol{\theta}_*$.

   In our context, to monitor a parameter vector which is identifiable, Castelloe and Zimmerman [13] proposed to mark a set of observations and follow the parameters to which these observations are allocated at the end of each sweep. This would allow us to overcome the label switching problem. The choice of these observations must be such that those that are expected to fluctuate across sweeps are likely to be selected.

2. Simulate $C > 1$ chains of equal length T with overdispersed starting values.

3. Choose a base batch size b, Brooks and Gelman [10] suggested for example $b \approx \frac{T}{20}$.

4. For $q = 1, \ldots, \frac{T}{20}$, compute $PSRF1^{(q)}(\theta_i)$, $PSRF2^{(q)}(\theta_i)$, $MPSRF1^{(q)}(\boldsymbol{\theta}_*)$ and $MPSRF2^{(q)}(\boldsymbol{\theta}_*)$.

5. Determine q_0 such that for $q > q_0$: (a) the plots for $PSRF1^{(q)}(\theta_i)$, $PSRF2^{(q)}(\theta_i)$, $MPSRF1^{(q)}(\boldsymbol{\theta}_*)$ and $MPSRF2^{(q)}(\boldsymbol{\theta}_*)$ are close to 1; (b) the plots for pairs of numerator and denominator for $PSRF1^{(q)}(\theta_i)$, $PSRF2^{(q)}(\theta_i)$, maximum eigenvalue of $Wm(\boldsymbol{\theta}_*)$ and maximum eigenvalue of $WmWc(\boldsymbol{\theta}_*)$ have settled approximately to a common value. The first $q_0 b$ observations could then be discarded and use the remaining ones used for inference.

6.2.2 Examples

We followed Castelloe and Zimmerman [13] to assess the convergence of the examples we have followed in chapters, we present the results for the Ruspini data set and the Lubischew's beetle data for both trans-dimensional samplers we have used: RJMCMC and BDMCMC.

Three chains were run for each example, from an overdispersed starting point. The first 200000 iterations were discarded as burn-in and the following 100000 were thinned every 50 iterations to end with a total $T = 2000$ sweeps for each chain. We selected $b = 100$ and evaluated the corresponding diagnostic statistics for each of the resulting 10 batches.

The number of components, k, was used as a "model" indicator and for each example eight observations were selected and the parameter vector $\boldsymbol{\theta}_*$ was formed by all the mean vectors to which each observation was allocated. The first two observations were selected by obtaining the minimum spanning tree and keeping the two observations that were joined by the largest edge in the tree. After removing this edge, we repeated the procedure again for the two resulting data subsets and kept the four observations that were joined with the largest edge in each tree. Finally, we selected the observations alternating the minimum and maximum with each dimension. We hope to select observations that exhibit a different behaviour across sweeps. That is we expect them to be either between two clusters that compete for the allocation of the observation or potential outliers. The selected observations for the Ruspini data set were: 1, 31, 48, 20, 17, 73, 43 and 44. For the Lubischew's beetle data set the selected observations

were: 56, 72, 26, 46, 53, 68, 25 and 79, see Figure 6.1.

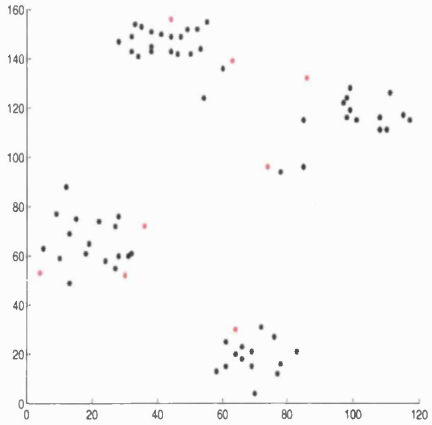


Figure 6.1: Ruspini data, selected points to monitor for convergence assessment are shown in red.

Ruspini data

The resulting plots show that for the Ruspini data set in the RJMCMC sampler, although the mean vector corresponding to observation 75 displayed some peaks in Figures 6.2(a) and 6.2(b), the values for the statistics are close to one after the seventh batch. In Figures 6.2(c) and 6.2(d) we can verify that the values for each pair are very close together. The maximum eigenvalues do perhaps take longer to settle to a common value, whereas the individual values showed to have settled to a common value after the seventh batch. The individual values corresponding to observations 1, 17 and 75 were the only ones displayed for clarity of presentation. The RJMCMC samplers used in Chapter 4 gave similar results for the MST split/combine move, despite observing a peak in Figure 6.5(b), the values are close to one after batch six. However, the plots in Figures 6.8(c) and 6.8(d), show that these values have not settled to a common value yet, giving evidence of non convergence for this sampler.

The resulting plots for the Ruspini data set in the BDMCMC sampler, show even larger peaks for the means corresponding to observation 75, see Figures 6.3(a) and 6.3(b). The values for the statistics are close to one after the seventh batch. Here, some of the values for $MPSRF1^{(q)}(\boldsymbol{\theta}_*)$ and $MPSRF2^{(q)}(\boldsymbol{\theta}_*)$ could not be calculated as the matrix $WmWc(\boldsymbol{\theta}_*)$ is close to singularity. In Figures 6.2(c) and 6.2(d) we observe that the values for each pair are very close together and have settled to a common value after the seventh batch.

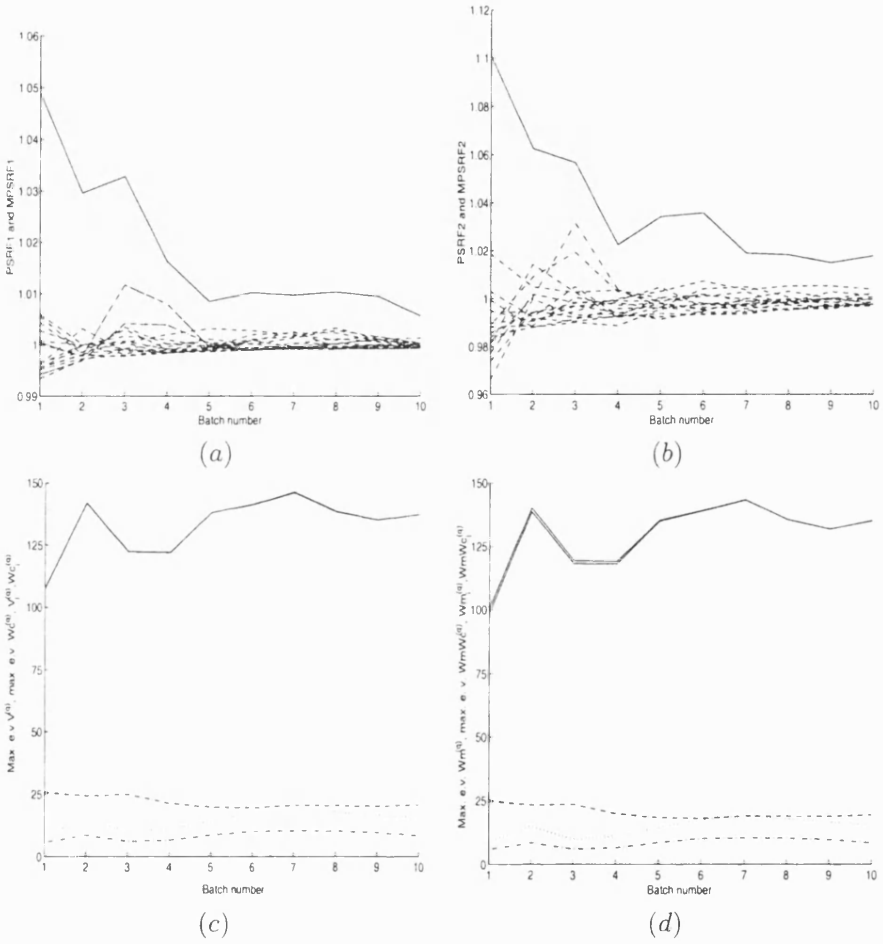


Figure 6.2: Ruspini data set, RJMCMC sampler. *(a)* Solid line: $MPSRF1^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $PSRF1^{(q)}(\theta_i)$ by batch number q. *(b)* Solid line: $MPSRF2^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $PSRF2^{(q)}(\theta_i)$ by batch number q. *(c)* Solid lines: Maximum eigenvalues for $\widehat{V}^{(q)}(\boldsymbol{\theta}_*)$ and $Wc^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $\widehat{V}^{(q)}(\theta_i)$ and $Wc^{(q)}(\theta_i)$ (for some observations) by batch number q. *(d)* Solid lines: Maximum eigenvalues for $Wm^{(q)}(\boldsymbol{\theta}_*)$ and $WmWc^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $Wm^{(q)}(\theta_i)$ and $WmWc^{(q)}(\theta_i)$ (for some observations) by batch number q.

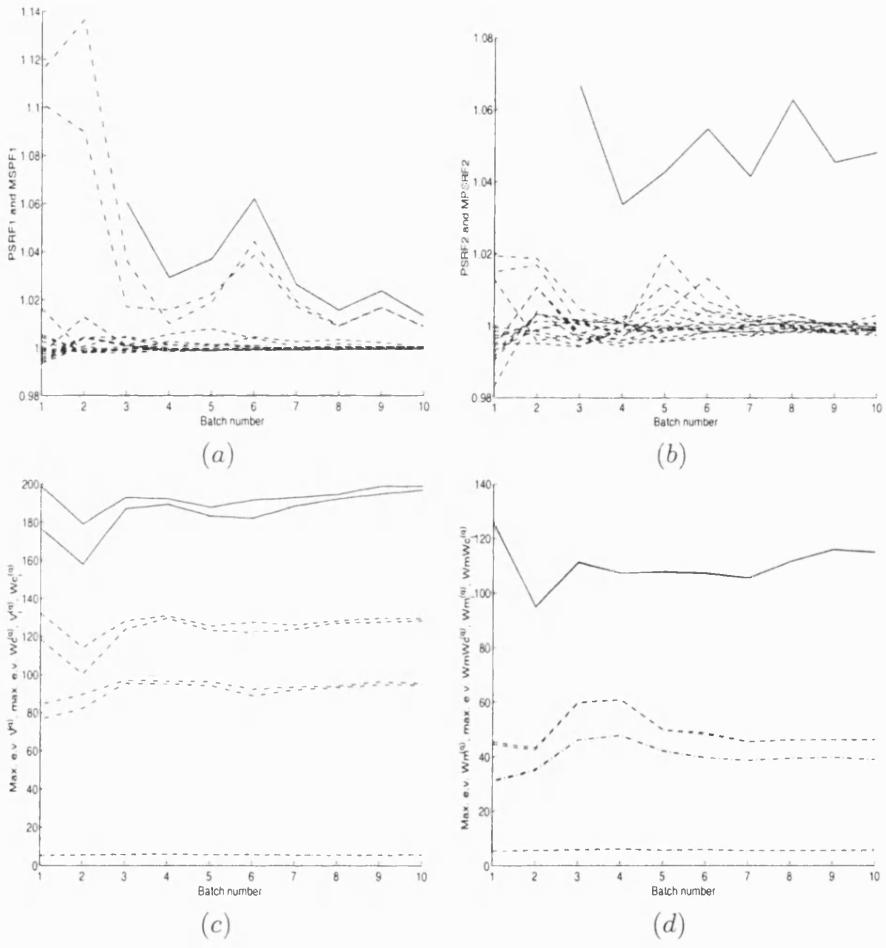


Figure 6.3: Ruspini data set, BDMCMC sampler. *(a)* Solid line: $MPSRF1^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $PSRF1^{(q)}(\theta_i)$ by batch number q. *(b)* Solid line: $MPSRF2^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $PSRF2^{(q)}(\theta_i)$ by batch number q. *(c)* Solid lines: Maximum eigenvalues for $\widehat{V}^{(q)}(\boldsymbol{\theta}_*)$ and $Wc^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $\widehat{V}^{(q)}(\theta_i)$ and $Wc^{(q)}(\theta_i)$ (for some observations) by batch number q. *(d)* Solid lines: Maximum eigenvalues for $Wm^{(q)}(\boldsymbol{\theta}_*)$ and $WmWc^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $Wm^{(q)}(\theta_i)$ and $WmWc^{(q)}(\theta_i)$ (for some observations) by batch number q.

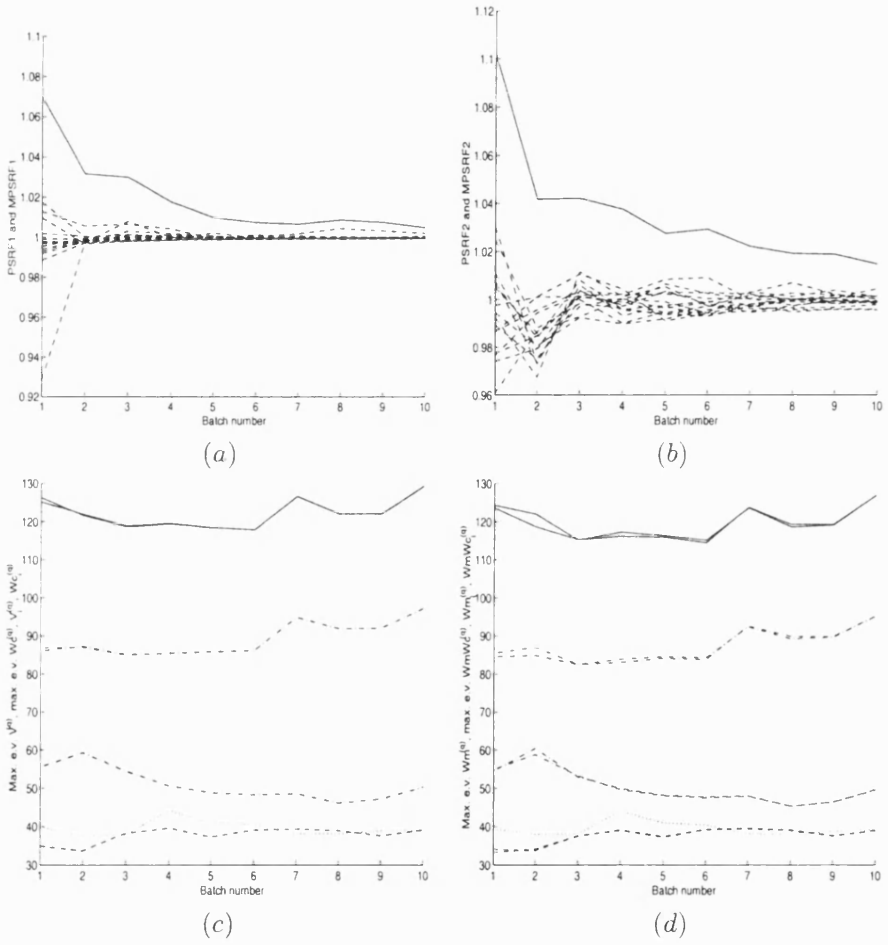


Figure 6.4: Ruspini data set, RJMCMC sampler PC split/combine move. *(a)* Solid line: $MPSRF1^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $PSRF1^{(q)}(\theta_i)$ by batch number q. *(b)* Solid line: $MPSRF2^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $PSRF2^{(q)}(\theta_i)$ by batch number q. *(c)* Solid lines: Maximum eigenvalues for $\widehat{V}^{(q)}(\boldsymbol{\theta}_*)$ and $Wc^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $\widehat{V}^{(q)}(\theta_i)$ and $Wc^{(q)}(\theta_i)$ (for some observations) by batch number q. *(d)* Solid lines: Maximum eigenvalues for $Wm^{(q)}(\boldsymbol{\theta}_*)$ and $WmWc^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $Wm^{(q)}(\theta_i)$ and $WmWc^{(q)}(\theta_i)$ (for some observations) by batch number q.

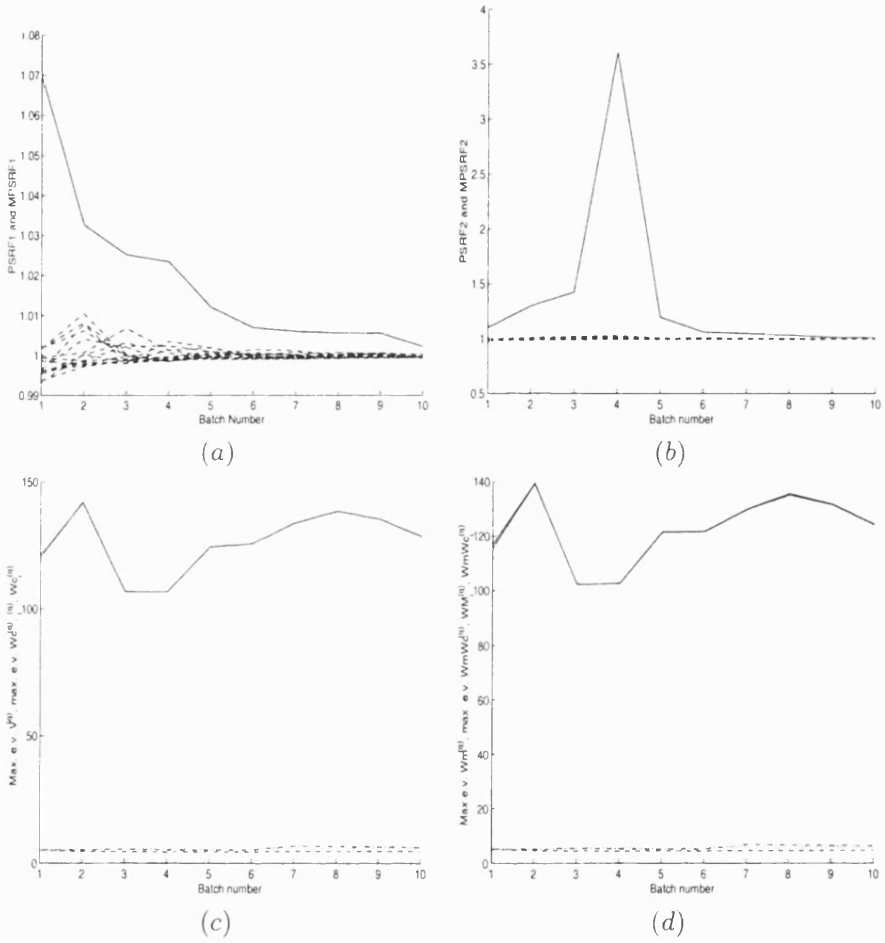


Figure 6.5: Ruspini data set, RJMCMC sampler MST split/combine move. *(a)* Solid line: $MPSRF1^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $PSRF1^{(q)}(\theta_i)$ by batch number q. *(b)* Solid line: $MPSRF2^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $PSRF2^{(q)}(\theta_i)$ by batch number q. *(c)* Solid lines: Maximum eigenvalues for $\widehat{V}^{(q)}(\boldsymbol{\theta}_*)$ and $Wc^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $\widehat{V}^{(q)}(\theta_i)$ and $Wc^{(q)}(\theta_i)$ (for some observations) by batch number q. *(d)* Solid lines: Maximum eigenvalues for $Wm^{(q)}(\boldsymbol{\theta}_*)$ and $WmWc^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $Wm^{(q)}(\theta_i)$ and $WmWc^{(q)}(\theta_i)$ (for some observations) by batch number q.

Lubischew's beetle data

For the Lubischew's beetle data none of the values for $MPSRF1^{(q)}(\boldsymbol{\theta}_*)$ and $MPSRF2^{(q)}(\boldsymbol{\theta}_*)$ could be computed. From the plots of the remaining statistics we observe that for the RJMCMC sampler, the values in Figures 6.6(a) and 6.6(b) are not close to one and the pairs of statistics in graphs 6.6(c) and 6.6(d) are not close together and have not settled to a common value for the first four batches. After this the sampler shows no evidence of lack of convergence.

The BDMCMC sampler for the Lubischew's beetle data we noticed that in the first batches that the last entry of the mean vector corresponding to observation 79, showed some peaks. Evidence of convergence in all plots in Figure 6.7 can be observed after batch seven.

To inspect the behaviour of the test in earlier stages of the runs, we ran only 100,000 iterations as a burn-in period and considered the following 100,000 iterations thinned every 50. We selected the Ruspini data set for the BDMCMC sampler and the Lubischew's beetle data for the RJMCMC. As we suspected, the BDMCMC sampler for the Ruspini data set showed evidence of convergence for a smaller number of iterations. The plots for the convergence assessment are shown in Figure 6.8. Notice that for this case, it could be argued that the maximum eigenvalues for the matrices corresponding to the pairs $(V^{(q)}, Wc^{(q)})$ and $(Wm^{(q)}, WmWc^{(q)})$, remain close together but could need a longer run to settle approximately to a common value.

However, the run for the Lubischew's beetle data with the RJMCMC sampler showed that more iterations are needed to have evidence of convergence. Results are shown in Figure 6.9, we can see that the values for the MPSRF1, PSRF1, MPSRF2, PSRF2 are not approximately one until the very last batch. Also the values for the maximum eigenvalues corresponding to the pairs $(V^{(q)}, Wc^{(q)})$ and $(Wm^{(q)}, WmWc^{(q)})$ are not close together and have not settled to a common value.

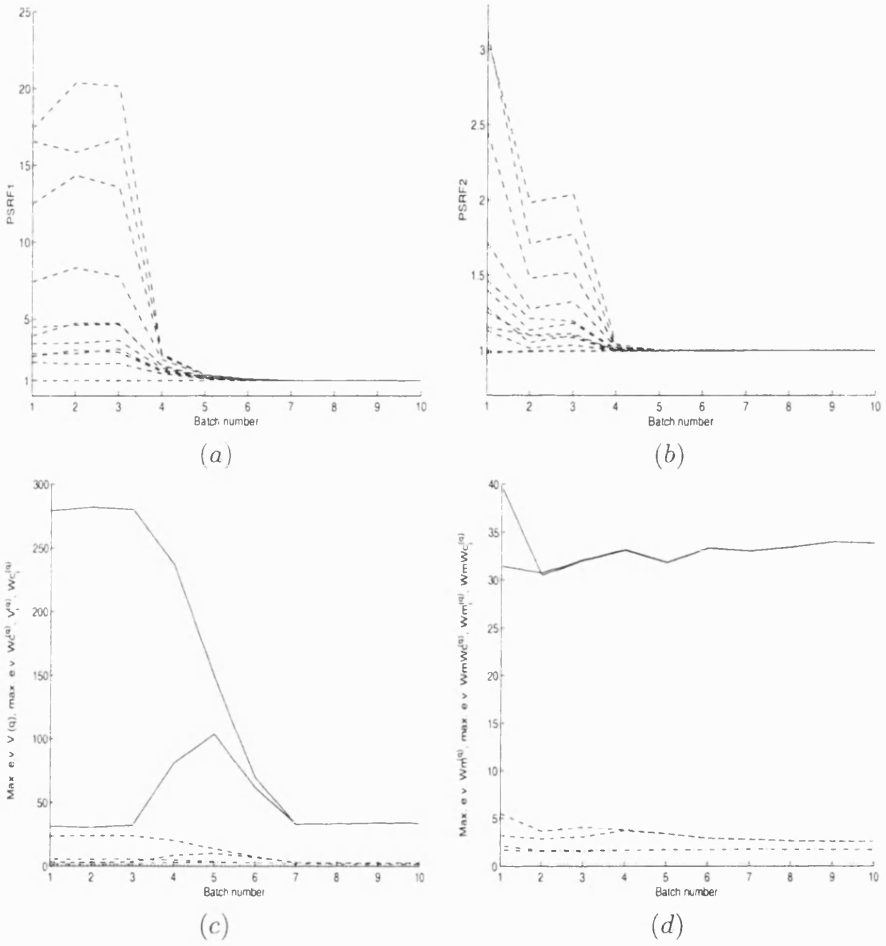


Figure 6.6: Lubischew's beetle data set, RJMCMC sampler. *(a)* $PSRF1^{(q)}(\theta_i)$ by batch number q. *(b)* $PSRF2^{(q)}(\theta_i)$ by batch number q. *(c)* Solid lines: Maximum eigenvalues for $\widehat{V}^{(q)}(\boldsymbol{\theta}_*)$ and $Wc^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $\widehat{V}^{(q)}(\theta_i)$ and $Wc^{(q)}(\theta_i)$ (for some observations) by batch number q. *(d)* Solid lines: Maximum eigenvalues for $Wm^{(q)}(\boldsymbol{\theta}_*)$ and $WmWc^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $Wm^{(q)}(\theta_i)$ and $WmWc^{(q)}(\theta_i)$ (for some observations) by batch number q.

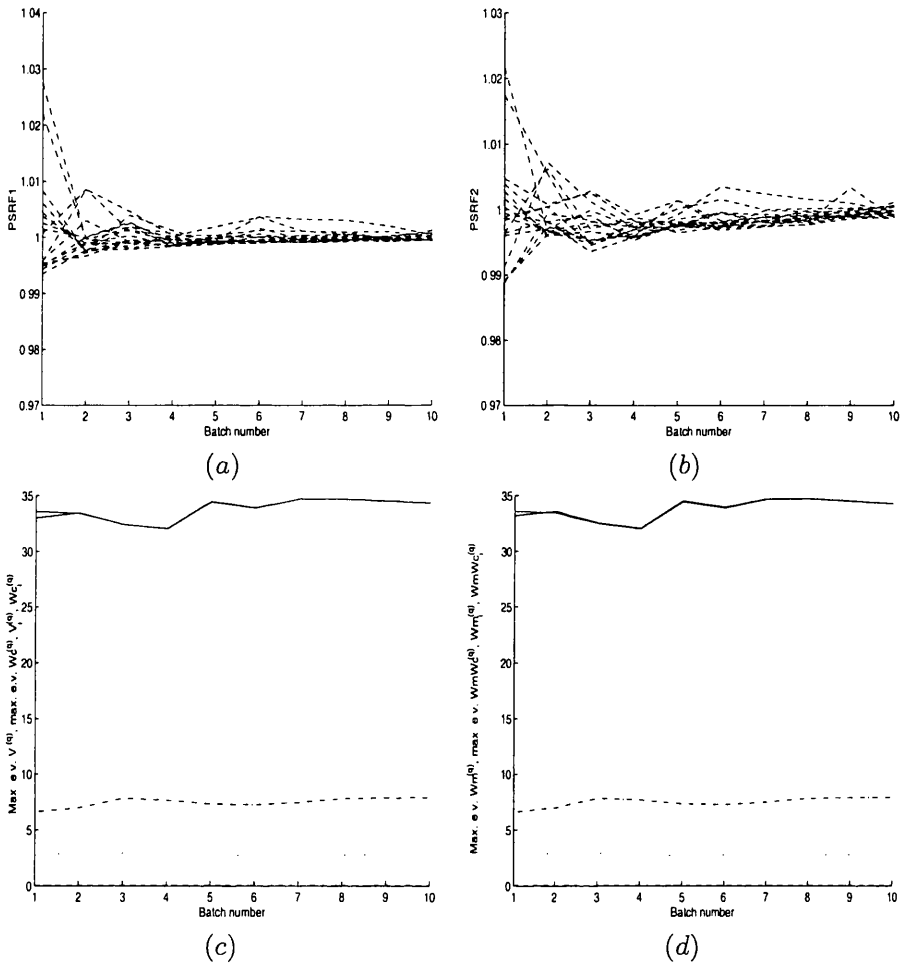


Figure 6.7: Lubischew's beetle data set, BDMCMC sampler. *(a)* $PSRF1^{(q)}(\theta_i)$ by batch number q. *(b)* $PSRF2^{(q)}(\theta_i)$ by batch number q. *(c)* Solid lines: Maximum eigenvalues for $\widehat{V}^{(q)}(\boldsymbol{\theta}_*)$ and $Wc^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $\widehat{V}^{(q)}(\theta_i)$ and $Wc^{(q)}(\theta_i)$ (for some observations) by batch number q. *(d)* Solid lines: Maximum eigenvalues for $Wm^{(q)}(\boldsymbol{\theta}_*)$ and $WmWc^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $Wm^{(q)}(\theta_i)$ and $WmWc^{(q)}(\theta_i)$ (for some observations) by batch number q.

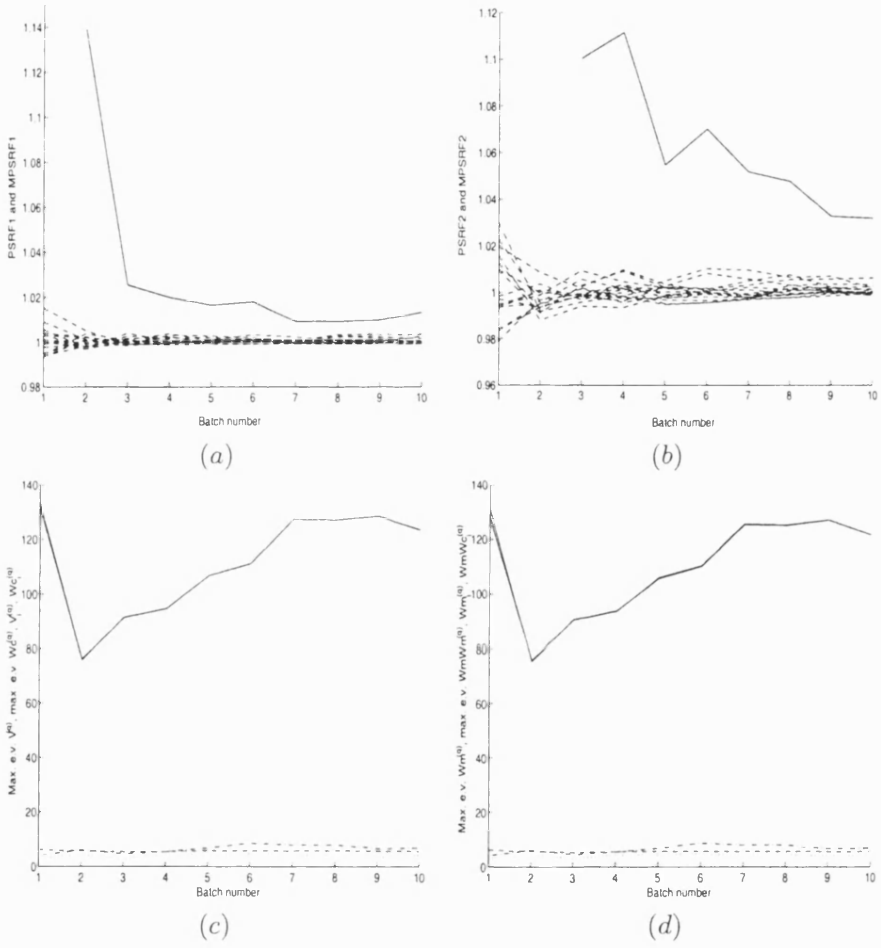


Figure 6.8: Ruspini data set, BDMCMC sampler for a burn-in period of 100000 iterations. *(a)* Solid line: $MPSRF1^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $PSRF1^{(q)}(\theta_i)$ by batch number q. *(b)* Solid line: $MPSRF2^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $PSRF2^{(q)}(\theta_i)$ by batch number q. *(c)* Solid lines: Maximum eigenvalues for $\widehat{V}^{(q)}(\boldsymbol{\theta}_*)$ and $Wc^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $\widehat{V}^{(q)}(\theta_i)$ and $Wc^{(q)}(\theta_i)$ (for some observations) by batch number q. *(d)* Solid lines: Maximum eigenvalues for $Wm^{(q)}(\boldsymbol{\theta}_*)$ and $WmWc^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $Wm^{(q)}(\theta_i)$ and $WmWc^{(q)}(\theta_i)$ (for some observations) by batch number q.

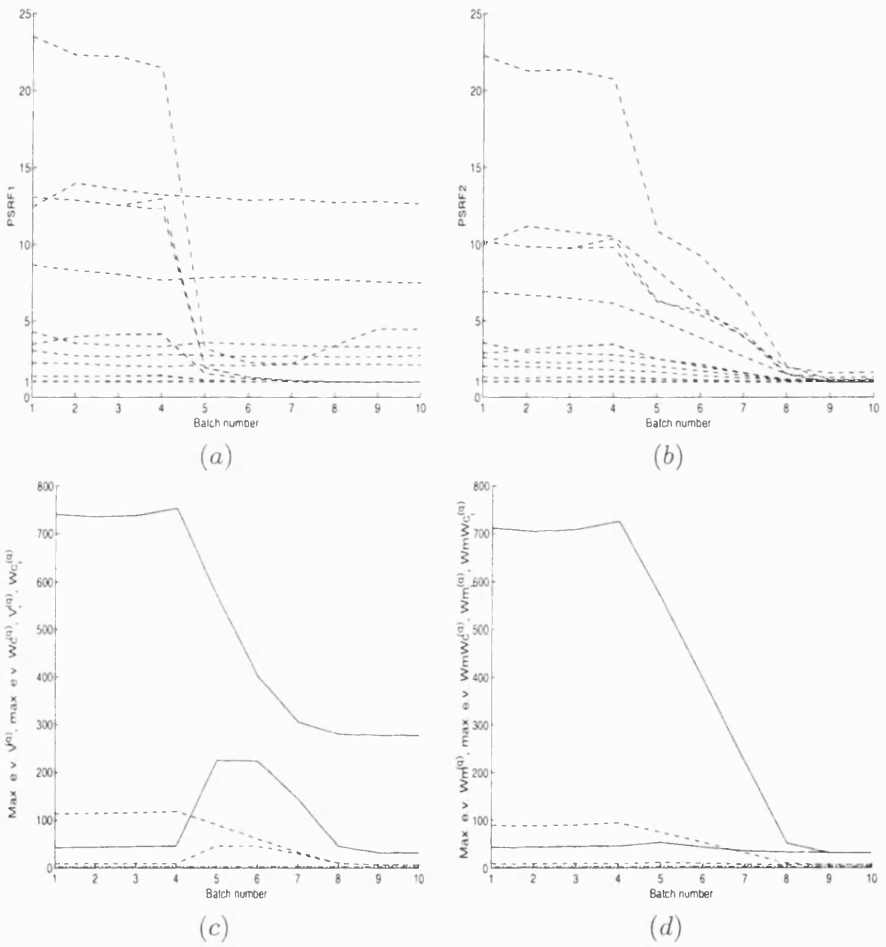


Figure 6.9: Lubischew's beetle data set, RJMCMC sampler for a burn-in period of 100000 iterations.. *(a)* $PSRF1^{(q)}(\theta_i)$ by batch number q. *(b)* $PSRF2^{(q)}(\theta_i)$ by batch number q. *(c)* Solid lines: Maximum eigenvalues for $\widehat{V}^{(q)}(\boldsymbol{\theta}_*)$ and $Wc^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $\widehat{V}^{(q)}(\theta_i)$ and $Wc^{(q)}(\theta_i)$ (for some observations) by batch number q. *(d)* Solid lines: Maximum eigenvalues for $Wm^{(q)}(\boldsymbol{\theta}_*)$ and $WmWc^{(q)}(\boldsymbol{\theta}_*)$ by batch number q. Dashed lines: $Wm^{(q)}(\theta_i)$ and $WmWc^{(q)}(\theta_i)$ (for some observations) by batch number q.

In general, we conclude that there is enough evidence to support the hypothesis that the chain has reached equilibrium for the long runs used in previous chapters. Trans-dimensional MCMC samplers may require longer runs to reach equilibrium than many MCMC samplers in fixed parameter spaces, particularly in high dimensional problems. Despite having considered a long burn-in period and a long set of iterations, we have seen that the first few hundred iterations still display some instability and could be discarded before carrying out inference.

CHAPTER 7

A more flexible model for a cluster

In practical applications, the description of one group by only one component of the mixture model may prove to be over ambitious. It may be inappropriate to assume that the data belong to the same group which is well described by a multivariate normal distribution, or in fact any other simple multivariate distribution. In two or three dimensional problems it may be possible to inspect the data in order to choose suitable parametric distributions to model clusters but this becomes increasingly unrealistic in higher dimensions. Stephens [59] points out that it would be useful to distinguish between the number of components in the model and the number of *groups* in the data in a context where defining groups is the main aim of the analysis. In this chapter we will consider a mixture of multivariate normal distributions with restricted covariance matrices as the underlying model for the cluster analysis. The fitted model will be used to define clusters as submixtures of components. We propose the use of two criteria in combination, this allows us to avoid merging pairs of components which swap observations only when the sampled values for the parameters of the components are in the tails of the corresponding distributions. They also indicate when a large proportion of observations is swapped between components which are very close together. This might help to identify some overlapping clusters in a more efficient way.

We begin with some remarks about the performance of multivariate normal distributions when faced with data that is inconsistent with such a model. We have often observed:

(i) that small weighted components are often included possibly to accommodate departures from normality including outliers.

(ii) that the fitted model frequently copes with non-normality by preferring a small number of highly dispersed components to a more complex model with a larger number of components

The intuitive idea we pursue in this chapter is to relax the assumption that one model-based component is used to describe a single cluster. Instead, we shall allow a submixture of such components to represent a cluster. At the same time, we are able to restrict the shapes of the multivariate normal distributions, allowing a simpler fitting to each element of a submixture. Thus we need to produce more, possibly many more model components than groups and subsequently find a method of combining some components into submixtures that describe sensible clusters.

The first requirement can be achieved by restricting the covariance structure of the multivariate normal components which also has the benefit of introducing simpler parametric forms for the basic element of our model.

For the second requirement, we shall consider two criteria for merging components into submixtures. The first is based on the closeness of the model component distributions and the second is based of the degree to which components dispute the ownership of the data.

In the next section we describe the behaviour of the two restricted models that we are considering and their effect on the posterior number of components. This will be crucial to our future conclusions using submixtures of components to carry out the cluster analysis. Our main objective is to obtain a fine partition of the existing groups into several components, particularly for data sets with overlapped groups, allowing the method to capture complex data structures in a more efficient way. Care is needed to avoid the extreme situation where one component is used to describe a single observation which belongs to a compact group.

7.1 Model I

Firstly, we consider a very simple model (model I), namely a mixture of spherical multivariate normal distributions. Let $\mathbf{y} = \mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n$ denote a data set, where $\mathbf{y}_j$ is a $p-$vector with probability density function given by the following equation:

$$f(\mathbf{y}_j;\Psi)=\sum_{i=1}^{k} w_i N_p(\mathbf{y}_j|\boldsymbol{\mu}_i,\tau_i^{-1}I_p), \tag{7.1}$$

where I_p denotes the p dimensional identity matrix.

The importance of allowing the volumes of the normal components to be different when considering the same shape and the same orientation has been pointed out by Celeux and Govaert [15]. They showed that these models are capable of detecting many clustering structures without needing complex algorithms. However, they only considered two dimensional data. Our main concern is to prevent the use of highly dispersed distributions misrepresenting the data. We allow the τ's to vary from component to component to preserve some flexibility in the model but place a tight restriction on their size through the prior distributions we assign, as we shall now describe.

Consider observations $\mathbf{y}_1,\ldots,\mathbf{y}_n$, where $\mathbf{y}_j$ has a distribution given in equation (7.1). The Bayesian hierarchical model has the following prior structure

$$\begin{aligned} w &\sim Dirichlet(\delta_1,\ldots,\delta_k),\\ \boldsymbol{\mu}_i &\sim N_p(\xi,\kappa^{-1}),\\ \tau_i &\sim gamma(\alpha,\alpha\beta), \end{aligned}$$

for $i=1,\ldots,k$. Where ξ is an $p\times 1$ vector, κ is a $p\times p$ matrix and δ_i, α and β are scalars.

The corresponding posterior full conditional distributions for the Gibbs sampler step in the trans-dimensional algorithm that will be used are

$$\begin{aligned} w &\sim Dirichlet(\delta_1+n_1,\ldots,\delta_k+n_k),\\ \tau_i|\cdots &\sim gamma(\alpha+(pn_i)/2,\alpha\beta+\frac{1}{2}\sum_{j=1}^{n_i}(\mathbf{y}_j-\boldsymbol{\mu}_i)'I_p(\mathbf{y}_j-\boldsymbol{\mu}_i)),\\ \boldsymbol{\mu}_i|\cdots &\sim N((\kappa+n_i\tau_i I_p)^{-1}(n_i\tau_i I_p\overline{\mathbf{y}}_i+\kappa\xi),(\kappa+n_i\tau_i I_p)^{-1}), \end{aligned}$$

where $n_i=\#\{j:z_j=i\}$ for the allocation variables z_j and $\overline{\mathbf{y}}_i=1/n_i\sum_{\{j:z_j=i\}}^{n_i}\mathbf{y}_j$.

In particular, we take ξ_j as the midpoint of the corresponding observed interval of variation. Let R_j denote the length of these intervals for $j=1,\ldots,p$, the matrix κ is

a diagonal matrix given as

$$\kappa = \begin{pmatrix} 1/R_1^2 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 1/R_p^2 \end{pmatrix}.$$

The values for the scalars are taken as follows: $\delta_i = 1$, $\alpha = c^2$ and $\beta = R_{max}/c$, where $R_{max} = \max\{R_1, \dots, R_p\}$. The constant c is chosen to restrict the size of the τ's. We wish to make a sensible choice of the parameters α and β to ensure that we induce the use of more components without heading for the extreme case where a lot of observations are isolated.

To determine the value of c we consider an initial analysis of the data set. This constant will give information on whether the groups in the data set exhibit important gaps or they are likely to overlap. We consider the projection of all data points onto the direction of the variable with the maximum observed range R_{max}. The projected data $(y_1', y_2', \dots, y_n')$ are used to find the largest gap between adjacent y_i''s, defining γ as the largest difference between adjacent y_i''s. When there are well separated groups in the data set, γ tends to be much larger than when the groups overlap.

When assigning a value to c, we need to take into account the value of γ. If the value given to c is too large, the model will be likely to use a component to describe a single observation, even when the observation is close to the center of a compact cluster, which is not useful for the analysis. When the value of c is too small, over dispersed spheres will be used, misrepresenting the data. We set 5 as a lower bound for c based in our empirical experience. The value of c should be large when γ is small and it should be small if γ is large.

We have inspected several values of c in the different examples that will be described below. We propose the use of the values of c given according to the following stepwise function

$$c = \begin{cases} 40 & \text{if} \quad \gamma < 1 \\ 20 & \text{if} \quad 1 \leq \gamma < 5 \\ 10 & \text{if} \quad 5 \leq \gamma < 10 \\ 5 & \text{if} \quad \gamma \geq 10 \end{cases} \tag{7.2}$$

These values have proved useful and small variations of these numbers did not

show significant changes in the posterior inference. We have also tried other functions in particular the function

$$c = 5 + \frac{1}{\gamma^3}, \tag{7.3}$$

which has a similar behaviour as function (7.2).

Using function (7.3) to determine the value of c changes the posterior distribution of the model parameters, in particular of the number of components, k, from the results obtained using function (7.2). However, the conclusions in terms of the clusters they define do not change. We do not claim these values to be optimal, further investigation should be carried out.

7.1.1 Examples

A sample from the joint posterior distribution of the model was obtained through a BDMCMC sampler with overall birth rate $\lambda_b = 1$, running the birth and death process for a fixed period $t_0 = 1$. We present the results for some examples described in previous chapters and an additional simulated data set that will help evaluate the performance of the method when the groups in the data overlap.

Later on we will describe criteria to define the submixtures that describe a group. These will rely on the stationarity of the chain. Taking into account the results on the convergence analysis given in Chapter 6, we will run the BDMCMC sampler for a burn-in period of 300000 iterations followed by a period of 100000 iterations, thinned every 50, which we will use for inference. We found no evidence that suggest that the sampler has not reached equilibrium. The poor mixing over the number of components, which we have discussed in previous chapters, is not considered as a negative outcome in this case as long as the number of components is sufficiently larger than the number of groups.

Example 1: Old Faithful data.

The sampler fits a nine-component mixture with the highest posterior probability for the Old Faithful data under model I, sampled values for the mean vectors of a nine-component mixture are in Figure 7.1 (c), from the plot it is possible to discern the nine groups, we display different colours only to show the label switching observed in the MCMC output. Plots for the number of components, k are given in (a) and (b).

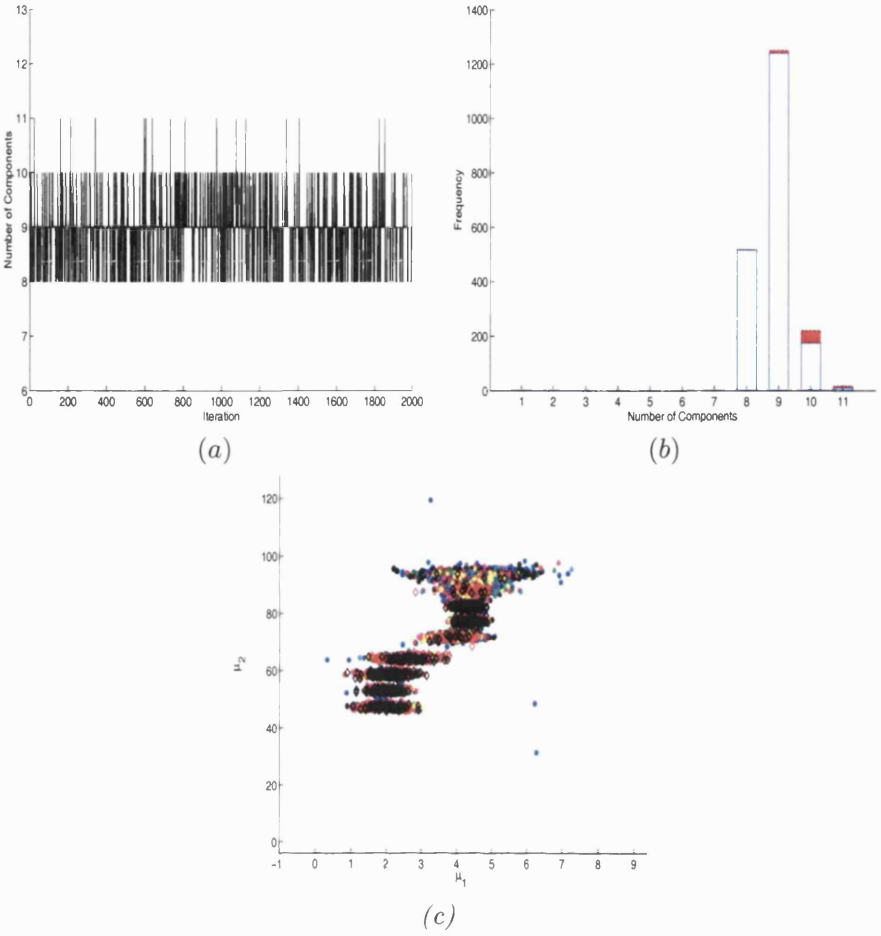


Figure 7.1: Model I: Old Faithful data. (a). Sampled values for the number of components k per iteration. (b) Barplot of the number of components. (c) Sampled values for the mean vectors for the 9-component mixture.

The groups we have identified in previous chapters are described by model I through several components. From the plot, we can see that the group of observations situated closer to the origin is possibly described by 3 or 4 components and the one away from the origin by 4 or 5 components. We will also obtain some information on the data between the two clearly defined groups to conclude if it may be considered a separate group or not.

Example 2: Ruspini data.

The sampler fits a five-component mixture with high probability to the Ruspini data set, sampled values for the mean vectors of the five-component mixture are shown in Figure 7.2 (c). The classification obtained from the dissimilarity matrix is consistent

with the results previously observed using the non-restricted models fitted with the BDMCMC sampler. In this case the model is fitting only one component per group, possibly including one group that might be reflecting non-normality.

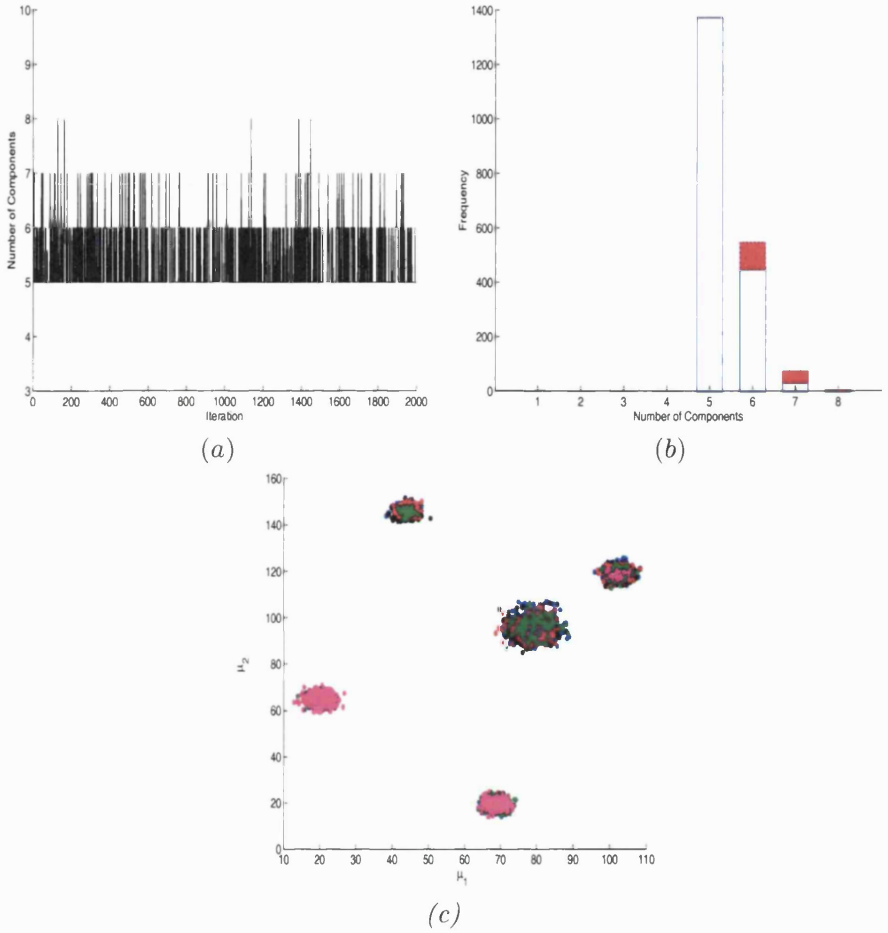
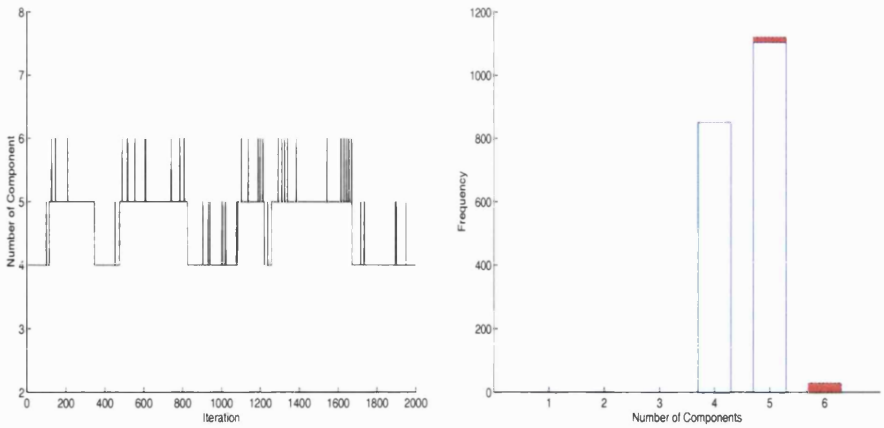


Figure 7.2: Model I: Ruspini data. (a). Sampled values for the number of components k per iteration. (b) Barplot of the number of components. (c) Sampled values for the mean vectors for the 5-component mixture.

Example 3: Iris data.

The Iris data set is described with a five-component mixture with the highest posterior probability, followed closely by a four-component mixture. The mixing over the number of components is not very good, results shown in Figure 7.3. The *setosa* species is described by only one component, the remaining components describe the *virginica* and *versicolor* species.

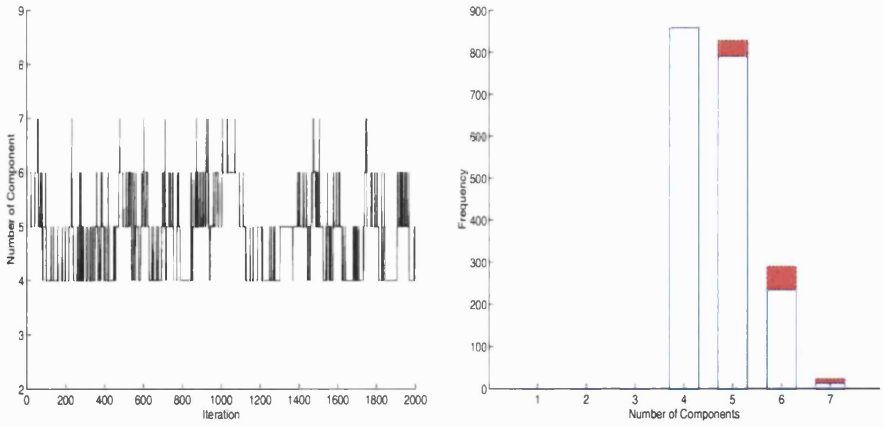


(a) (b)

Figure 7.3: Model I: Iris data. (a). Sampled values for the number of components k per iteration. (b) Barplot of the number of components.

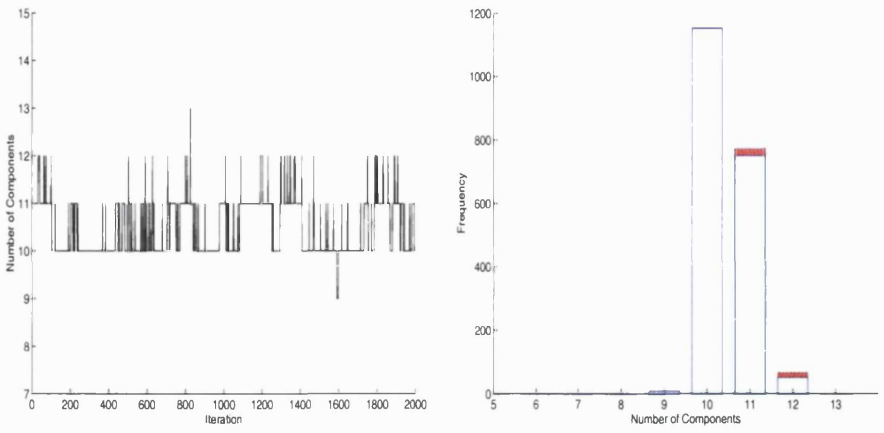
Example 4: Lubischew's beetle data.

The sampler for the Lubischew's beetle data mixes well over the number of components, see Figure 7.4. The posterior probabilities for four and five component mixtures are 0.4295 and 0.414 respectively. In the four-component mixture, the *heikertingeri* species is described with two components and in the five-component case with three components.



(a) (b)

Figure 7.4: Model I: Lubischew's beetle data. (a). Sampled values for the number of components k per iteration. (b) Barplot of the number of components.



(a) (b)

Figure 7.5: Model I: Simulated data. (a). Sampled values for the number of components k per iteration. (b) Barplot of the number of components.

Example 5: Simulated data.

The simulated data for this example is used to test the efficiency and flexibility of the methodology we discuss in this chapter. The data set contains 500 3-dimensional observations, which belong to six different groups of sizes $n_1 = 84$, $n_2 = 92$, $n_3 = 136$, $n_4 = 24$, $n_5 = 46$ and $n_6 = 118$ respectively. The observations were sampled uniformly in balls centered in $(9, 10, 10)^T$, $(10, 9, 10)^T$, $(10, 10, 9)^T$, $(11, 10, 10)^T$, $(10, 11, 10)^T$ and $(10, 10, 11)^T$ with weights 0.17, 0.18, 0.27, 0.05, 0.09 and 0.24. The radius of the balls was chosen as $1/\sqrt{2}$ so that they all " touch " each other. We expect that different components of the fitted mixture of multivariate normal distributions will dispute the ownership of the observations close to the borders of the balls. We therefore can assess the performance of the method when the fitted components overlap.

The unrestricted model using RJMCMC fits a seven-component mixture with the highest posterior probability. The classification obtained from a dissimilarity matrix based on the proportion of iterations that every pair of observations is allocated in the same component. The resulting groups have $(26, 47, 87, 100, 119, 64, 57)$. The BDMCMC sampler fits an eight-component mixture with the highest posterior probability. The classification based on the dissimilarity matrix splits the observations in eight groups with $(26, 44, 83, 70, 73, 74, 67, 63)$ observations respectively.

Using model I to fit a mixture of multivariate spherical normals to this simulated data set, a ten-component mixture has the highest posterior probability.

When using model I to describe the data sets described above, for the values of c given by function (7.2), the BDMCMC sampler fits a fine partition of the groups we obtained in previous chapters particularly when the density of the groups differs considerably from a spherical multivariate normal distribution. This results will be useful to describe a group as a submixture of the resulting fitted components.

Before describing the proposed criteria to distinguish submixtures of components that are likely to describe a single group, we consider a second model. We look for a slightly more flexible model where the different dispersion of variables in different directions will be considered and to compare the results with the ones obtained for model I.

7.2 Model II

In some cases one might be interested in fitting less components of a slightly different shape. If a compact group is well separated from the rest but has a density too different from a spherical normal, we might like the model to allow for an elliptical component to describe this group.

We consider the model given by

$$f(\mathbf{y}_j;\Psi) = \sum_{i=1}^{g} w_i N_d(\mathbf{y}_j|\boldsymbol{\mu}_i, \mathcal{D}_i), \tag{7.4}$$

where $\mathcal{D}_i = diag(\tau_{i1}^{-1}, \ldots, \tau_{ip}^{-1})$, with the following prior structure:

$$\begin{aligned} w &\sim D(\delta_1, \ldots, \delta_k), \\ \boldsymbol{\mu}_i &\sim N(\xi, \kappa^{-1}), \\ \tau_{il} &\sim gamma(\alpha, \alpha\beta_l), \end{aligned}$$

where $\delta_i = 1$, $\alpha = c^2$ and $\beta_l = R_l/c$.

The full conditional posterior distributions for the Gibbs step in the algorithm are

given as follows

$$\begin{aligned} w &\sim D(\delta_1 + n_1, \ldots, \delta_k + n_k), \\ \tau_{il}|\cdots &\sim gamma(\alpha + n_i/2, \alpha\beta_l + \frac{1}{2}\sum_{j:z_j=i}(y_{jl} - \mu_{i_l})^2) \\ \boldsymbol{\mu}_i|\cdots &\sim N((\kappa + n_i\mathcal{D}_i)^{-1}(n_i\mathcal{D}_i\overline{\mathbf{y}}_i + \kappa\xi), (\kappa + n_i\mathcal{D})^{-1}). \end{aligned}$$

where $l = 1, \ldots, p$, $n_i = \#\{j : z_j = i\}$ for the allocation variables z_j and $\overline{\mathbf{y}}_i = 1/n_i \sum_{\{j:z_j=i\}}^{n_i} \mathbf{y}_j$.

In this case, the values assigned to c need to be determined taking into account the value of γ, but less flexibility is needed as the model is allowing for variation in all directions, that is the range in each direction is explicitly considered. Smaller values are used for c compared to those used in model I. To fit the examples described below we use the stepwise function given as

$$c = \begin{cases} 20 & \text{if} \quad \gamma < 1 \\ 10 & \text{if} \quad 1 \leq \gamma < 5 \\ 5 & \text{if} \quad 5 \leq \gamma < 10 \\ 2.5 & \text{if} \quad \gamma \geq 10 \end{cases} \tag{7.5}$$

Small variations of the function (7.5) do not change the fitted model. We have also explored other functions that behave in a similar way, for example the function

$$c = 2.5 + \frac{1}{\gamma^2}. \tag{7.6}$$

Once again, the fitted model obtained by determining the value of c through function (7.6) is different form the model fitted by determining c using function (7.5), however, the conclusions in terms of the clusters so defined do not change.

7.2.1 Examples

We now describe the results obtained for the examples with the model described above. Again a sample from the joint posterior distribution of the model was obtained using a BDMCMC sampler with overall birth rate $\lambda_b = 1$, running the birth and death process for a fixed period $t_0 = 1$. We considered in this case a burn-in period of 400000 iterations followed by a period of 100000 iterations, thinned every 50, which we will use

for inference. For this model, preliminary results showed that a longer run was needed to ensure that the sampler had reached equilibrium.

Example 1: Old Faithful data.

The sampled values for the mean vectors of a seven-component mixture are shown in Figure 7.6 (c) Old Faithful data set, again different colours are only displayed to show the label switching observed in the MCMC output. This is the number of components with the highest posterior probability. Model II uses less components to describe the data than model I. In this case, the clearly defined groups are described by two components each and the fifth component is used to represent the data between these groups with a diagonal component.

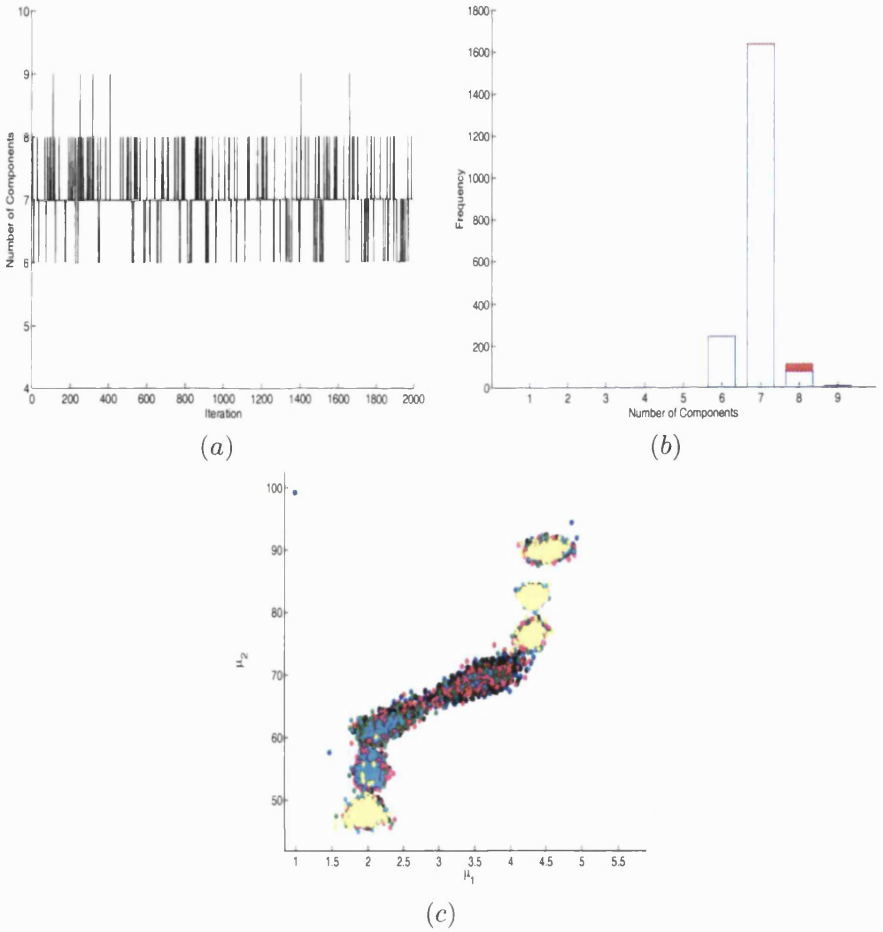


Figure 7.6: Model II: Old Faithful data. (a). Sampled values for the number of components k per iteration. (b) Barplot of the number of components. (c) Sampled values for the mean vectors for the 7-component mixture.

Example 2: Ruspini data.

The sampler fits a five-component mixture with the highest posterior probability to the Ruspini data set, results are shown in Figure 7.7. The classification obtained from the dissimilarity matrix is consistent with the results previously observed and we would expect that the criteria considered to define the groups as a submixture of components includes only one component per group for most groups in this example.

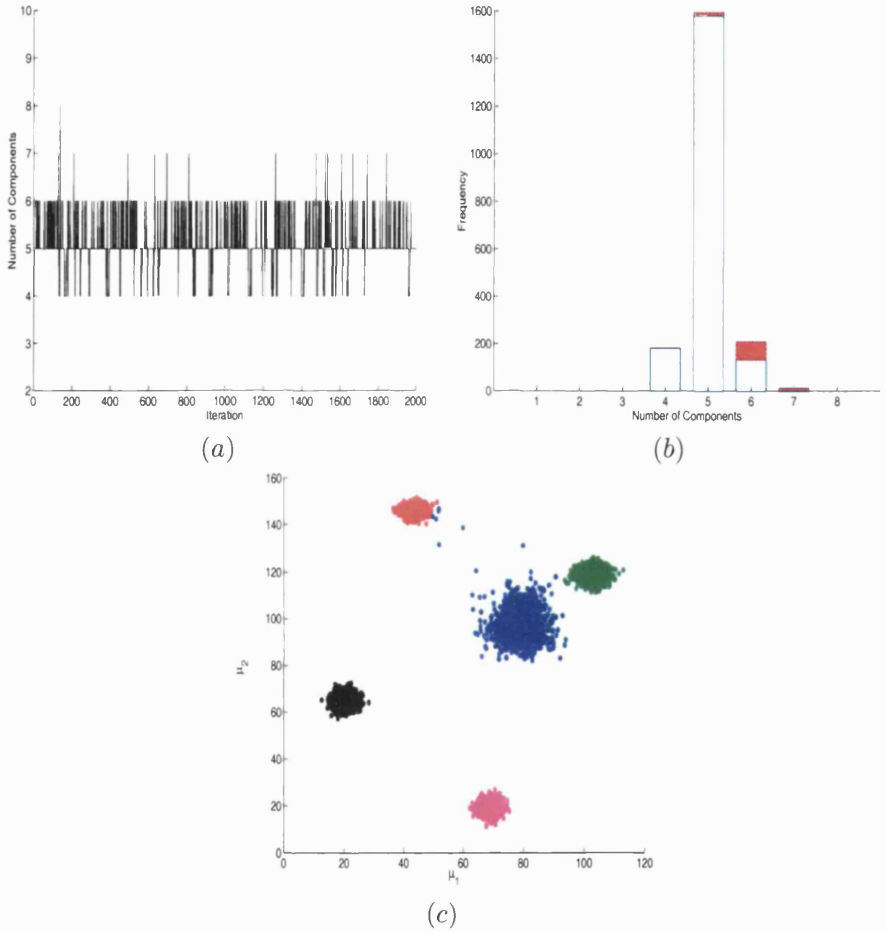
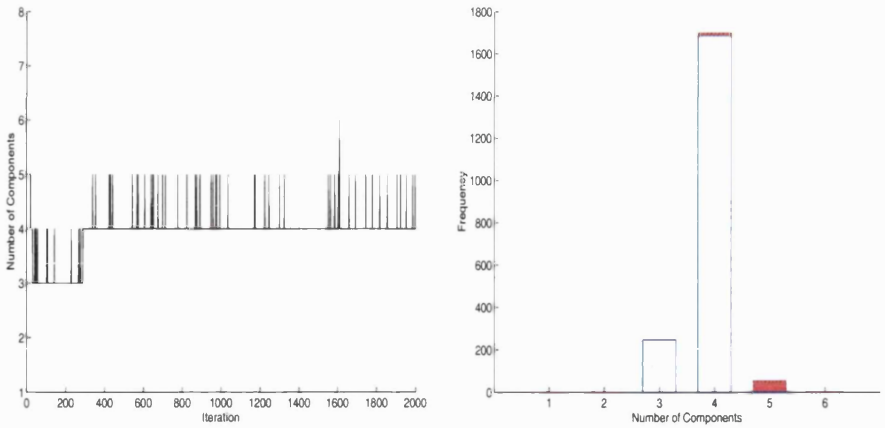


Figure 7.7: Model II: Ruspini data. (a). Sampled values for the number of components k per iteration. (b) Barplot of the number of components. (c) Sampled values for the mean vectors for the 5-component mixture.

Example 3: Iris data.

The Iris data set is described with a four-component mixture, results shown in Figure 7.8. The first species, the *setosa*, is described by one component and the other three are used to describe the data in the *virginica* and *versicolor* species.

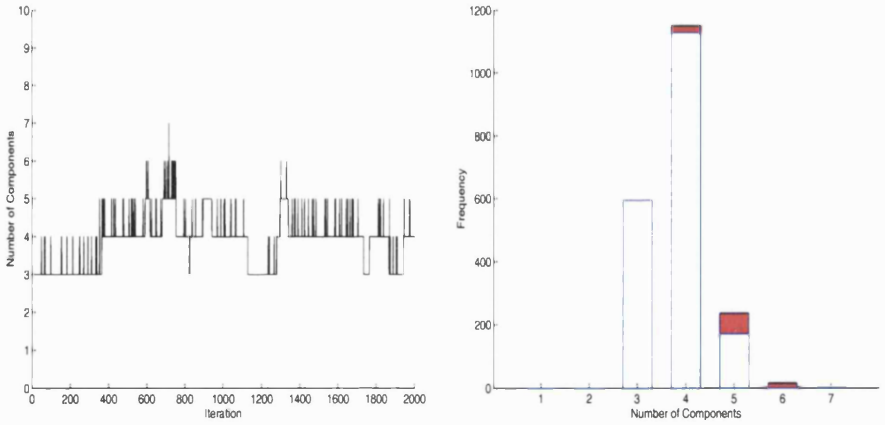


(a) (b)

Figure 7.8: Model II: Iris data. (a). Sampled values for the number of components k per iteration. (b) Barplot of the number of components.

Example 4: Lubischew's beetle data.

The sampler for the Lubischew's beetle data mixes poorly over the number of components, see Figure 7.9, and a four-component mixture has the highest posterior probability. Using model II, the *heikertingeri* species is describe with two components and the *concinna* and *heptapotamica* species with only one component each.

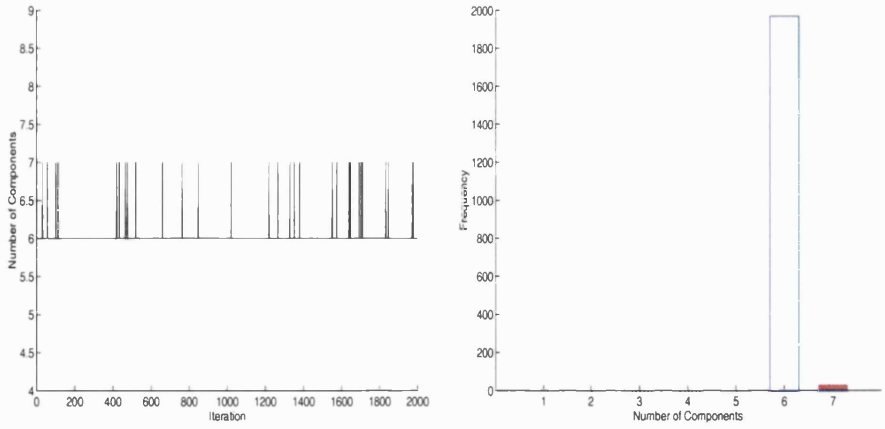


(a) (b)

Figure 7.9: Model II: Lubischew's beetle data. (a). Sampled values for the number of components k per iteration. (b) Barplot of the number of components.

Example 5: Simulated data.

The simulated data is described with a six-component mixture with the highest posterior probability. The number of components mixes poorly here, but the sampled



(a) (b)

Figure 7.10: Model II: Simulated data. (a). Sampled values for the number of components k per iteration. (b) Barplot of the number of components.

values for the mean vectors are close to the values used to simulate the data set.

Results for the two models described above can be used to explore the possibility to describe groups using a submixture of components. Posterior distributions for the component parameters are needed to decide if a component is a group on its own or if it describes a group together with other existing components.

To define the submixture of components that describes a group we need to identify the component parameters, in other words we need to remove the label switching. In the following section we discuss a way to do this using the early iterations of the sampler, before label switching occurs.

7.3 Identifying the component parameters in the mixture model

As we have described in section 1.3, the invariance of the likelihood under the relabelling of the components of a mixture model results in a difficulty in identifying the component parameters. We mentioned several methods proposed in the literature in recent years to deal with the label switching.

In particular, following Stephens [60], a simple procedure was proposed by Celeux [14] and used later in Celeux et al [16] to identify one modal region and estimate the component parameters, in particular we concentrate on the component means. The method selects one modal region using the early iterations of the Markov chain Monte

Carlo sampler. The choice of the initial set of iterations is not highly sensitive but should be enough to ensure that the resulting estimates are a reasonable approximation of the posterior means and before the label switching occurs. The component labels corresponding to the following iterations are permuted according to a k-means-type algorithm to select the permutation that is closest to the current set of means as described below.

Following this procedure, we post-processed the output of the trans-dimensional samplers for models I and II. We consider the sequence of d-dimensional vector samples of size N, conditional on the number of components k, $\psi^1, \ldots, \psi^m$, where $d = kp$, $\psi^i = (\mu_{1,i}, \ldots, \mu_{p,i}, \ldots, \mu_{k,i}, \ldots, \mu_{k,p})$ and m is the longest period observed before the label switching occurs. Here we are only taking into account iterations without empty components. We focused on the mean values because they have shown to be the most stable parameter which is covered rapidly by the sampler. However, results for the other parameters will be computed to verify results are consistent. The length of period m is determined in practice by looking at the number of observations allocated to each component. The period m is the period before the number of components allocated to each component has changed in more that one observation for all components.

Initial reference centers for $j = 1, \ldots, d$ are defined as

$$\overline{\psi}_j = \frac{1}{m} \sum_{i=1}^{m} \psi_j^i,$$

with corresponding variances

$$s_i = \frac{1}{m} \sum_{i=1}^{m} (\psi_j^i - \overline{\psi}_j)^2,$$

We denote $s_i^{[0]} = s_i$ for $i = 1, \ldots, d$. If we take $\overline{\psi}^{[0]} = \overline{\psi}$, the other $(k-1)$ centers can be deduced by permuting the labelling of the mixture components. The rth iteration is relabelled with the permutation j that minimises the normalized square distance,

$$\|\psi_j^{m+r} - \overline{\psi}^{[r-1]}\| = \sum_{i=1}^{d} \frac{(\psi_j^{m+r} - \overline{\psi}_i^{[r-1]})^2}{s_i^{[r-1]}},$$

for $j = 1, \ldots, k$ and $r = 1, \ldots, N - m$.

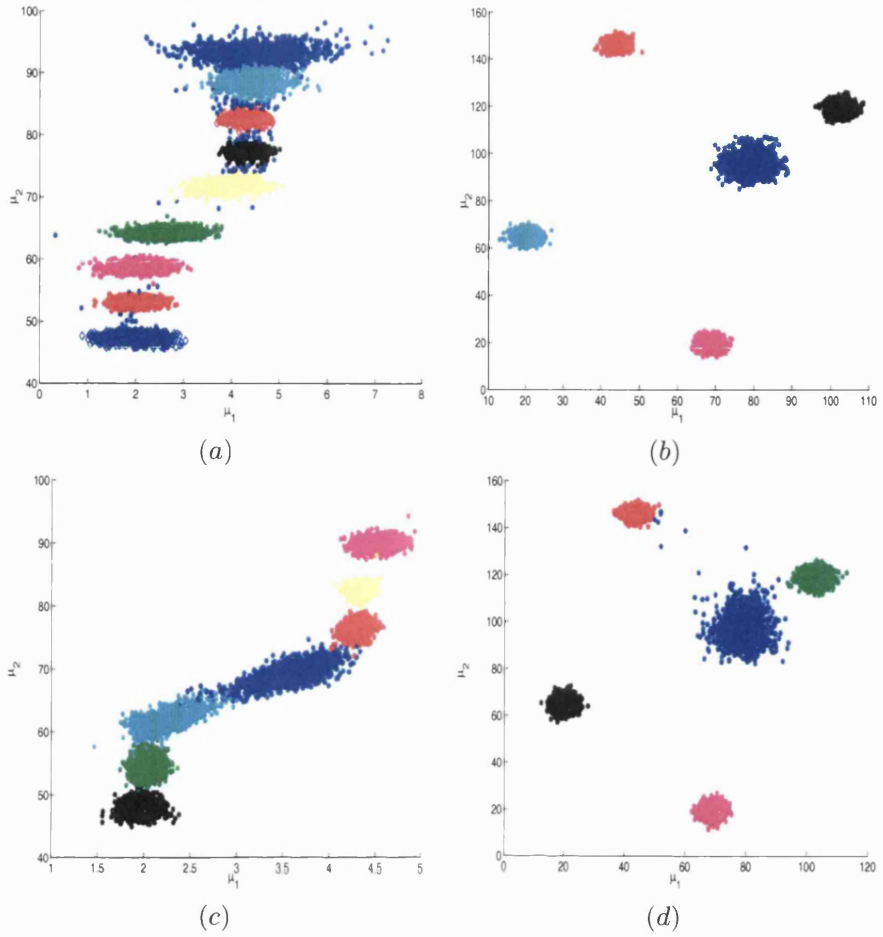


Figure 7.11: Sampled values for the mean vectors after removing the label switching: (a) Old Faithful data for a 9-component mixture, model I. (b) Ruspini data for a 5-component mixture, model I . (c) Old Faithful data for a 7-component mixture, model II. (d) Ruspini data for a 5-component mixture, model II.

The centers and normalising coefficients are updated after each iteration so

$$\begin{aligned}
\overline{\psi}^{[r]} &= \frac{m+r-1}{m+r}\,\overline{\psi}^{[r-1]} + \frac{1}{m+r}\,\psi^{m+r} \\
s_i^{[r]} &= \frac{m+r-1}{m+r}\,s_i^{[r-1]} + \frac{m+r-1}{m+r}\,(\overline{\psi}_i^{[r-1]} - \overline{\psi}_i^{[r]})^2 \\
&\quad + \frac{1}{m+r}\,(\psi_i^{m+r} - \overline{\psi}_i^{[r]})^2.
\end{aligned}$$

We have found the procedure to be efficient and helpful, it allows us to identify posterior component parameters for the fitted mixture so that their values can be used to determine the submixture of components that describes a group. If the mixture model we are analysing has more than 10 components it might be more efficient to

look at a smaller set of the permutations.

The performance of the method is presented for the two dimensional examples. The sampled mean vectors for the restricted models I and II for the Old Faithful and the Ruspini data sets are shown in Figure 7.11 after removing the label switching. It is also possible to identify the covariance matrix associated with each component as well as its mixing proportion, for each iteration. Results obtained by basing the identification procedure on the distances for the mixing proportions gave the same output for these examples.

After removing the label switching, we are now in a position to propose criteria to determine whether one or several components belong to the same group.

7.4 Clusters as a submixture of components

Consider the output of the BDMCMC samplers for the restricted models presented in sections 7.1 and 7.2. We are interested in criteria to indicate which of the fitted components might be merged to form a submixture that represents the same group. We propose two criteria: one will consider the proportion of allocated data that are *swapped* between components throughout the sampler and the other will give information on the distance between the densities of the components based on the *affinity* .

Broadly speaking, we are interested in measuring the distance between components. If a group is described by more than one component we expect their distributions to be " close ". Now, when the distance between two components is small, we want to learn about the proportion of observations whose ownership is disputed by the two components to which they are allocated. If this proportion is large, then the components are more likely to be describing one cluster. We suggest that a possible way to obtain straightforward information on the proportion of disputed observations is simply to look at how much data is swapped between components from iteration to iteration. We suggest to merge two components into one group if the proportion of observations exchanged between these components is large and the distance between the components is small, assessed by the affinity between two components. Using these criteria in combination gives information on particular situations. First it helps detecting when a pair of components swaps observations when values on the tails of the distributions of the component parameters are sampled.

Computations to obtain the proportion of swapped observations and the affinity between pairs of component will be made conditional on a given value of k, the number of components. The output of the trans-dimensional sampler will be post-processed. We expect results based on different number of components, which have similar and high posterior probability, to be consistent in terms of the groups they define. Once a value of k is fixed, the label switching is removed, identifying values for the parameters of each component and corresponding allocation vector for the data at each iteration.

7.4.1 Proportion of observations swapped between components

For the computation of the swaps, consider the allocation variable $Z_{i,j}$ for observation $i = 1, \ldots, n$ at iteration $j = 1, \ldots, J$, which takes a value $k = 1, \ldots, k_j$. To deal with the label switching we have extracted all the cases for $k_j = k$, that is, the calculation is done with an output that includes all the variables for a fixed value of k. Suppose that there are $R \leq J$ of such iterations.

We then consider the allocation variable Z_{ir} for observation $i = 1, \ldots, n$ at iteration $r = 1, \ldots, R$, which takes a value in $(1, \ldots, k)$. We compare it with the value for the same observation i but in the next iteration $r + 1$. A $k \times k$ matrix C is initialized to zeros and for $h = 1, \ldots, k$ and $l = h + 1, \ldots, k$, we add $1/(n_{h,r} + n_{l,r})$ to the element $C_{(h,l)}$, if either $Z_{i,(r+1)} = h$ and $Z_{i,r} = l$ or $Z_{i,(r+1)} = l$ and $Z_{i,r} = h$, where $n_{h,l}$ and $n_{l,r}$ are the number of observations allocated to components h and l at iteration r. Once the matrix is computed for all r, we divide it by R to get an approximate value of the proportion of observations that were exchanged between components.

7.4.2 The affinity between components

The Bhattacharyya distance [6] also known as the affinity is used as a measure of similarity between two probability distributions. The affinity between distributions P_1 and P_2 with corresponding densities p_1 and p_2 is defined as

$$A(p_1, p_2) = \int_{\mathbf{y}} \sqrt{p_1}\sqrt{p_2}\, d\mathbf{y}$$

The affinity is related to the Hellinger distance between two distributions P_1 and

P_2, with corresponding densities p_1, p_2 which is given as

$$H(P_1, P_2) = \left\{ \int_{\mathbf{y}} (\sqrt{p_1} - \sqrt{p_2})^2 \, d\mathbf{y} \right\}^{\frac{1}{2}} \tag{7.7}$$

Consider now, H^2

$$H(P_1, P_2)^2 = 2 - 2 \int_{\mathbf{y}} \sqrt{p_1}\sqrt{p_2} \, d\mathbf{y} \tag{7.8}$$

Hence, $H = \sqrt{2 - 2A(p_1, p_2)}$, where $A(p_1, p_2)$ is the affinity between P_1 and P_2.

The affinity between two multivariate normal distributions, $N(\mu_1, \Sigma_1)$ and $N(\mu_2, \Sigma_2)$ is obtained in Appendix B.

In the particular case where Σ_1 and Σ_2 are diagonal matrices, the affinity between p_1 and p_2 is given by

$$A(p_1, p_2) = \prod_{i=1}^{p} \left(\frac{2\sigma_{1,i}\sigma_{2,i}}{\sigma_{1,i}^2 + \sigma_{2,i}^2} \right)^{\frac{1}{2}} \exp\left\{ -\frac{1}{4} \sum_{i=1}^{p} \frac{(\mu_{1,i} - \mu_{2,i})^2}{\sigma_{1,i}^2 + \sigma_{2,i}^2} \right\}.$$

We compute the average affinity between all pairs of components. We denote as A the $k \times k$ matrix that shows the average value of the affinity for each pair of the k components, we will refer to it as the average affinity matrix. This will be done for all iterations of the sampler where the specified value of k is observed. From numerical experiments for model I and II, we found that in general the affinity matrix, A, has small values.

The behaviour of the swap and the affinity matrices will be discussed in section 7.5. We monitored 2000 iterations which came from 100000 iterations thinned every 50. We are interested on finding out if, from the components that are *close* together, there are pairs of components which are exchanging a large proportion of the observations allocated to these components.

7.5 Performance of criteria to identify clusters

In this section the affinity and swap matrices will be computed for the examples analysed in sections 7.1 and 7.2. We will consider the restricted models I and II and

will monitor the behaviour of these matrices and the results they give in terms of the description of the clusters.

7.5.1 Examples

Example 1: Old Faithful data.

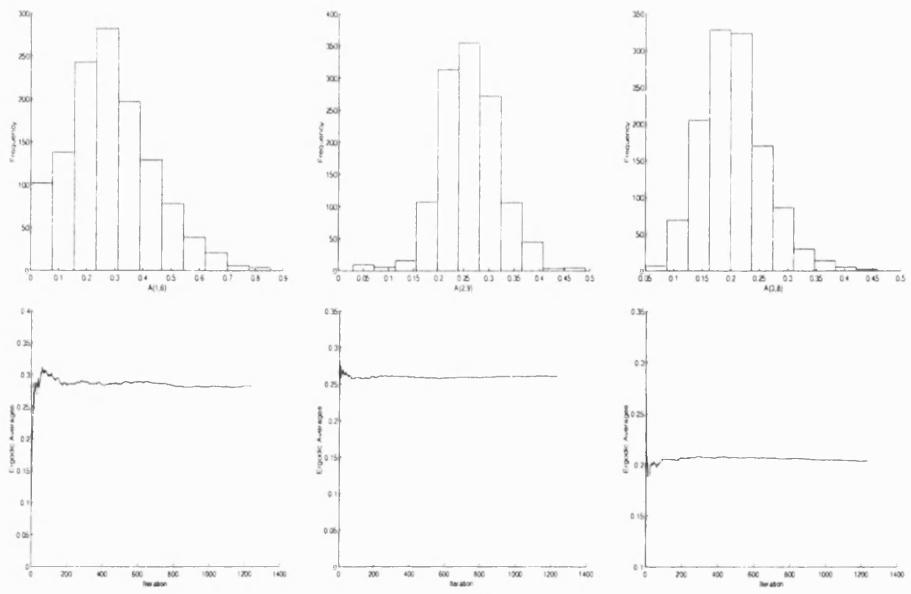
For the Old Faithful data we considered the values of k that had the highest posterior probability for models I and II, in both cases the other values of k had a much smaller posterior probability.

Model I

We begin with the results for the nine-component mixture fitted for model I. The matrix A below corresponds to the average affinity matrix, some of the values are too small and appear as zeros. The closer pairs of densities are those whose affinity is closer to 1.

$$A = \begin{pmatrix} 0 & 0.0270 & 0.0088 & 0.0032 & 0.0143 & 0.2821 & 0.0032 & 0.0063 & 0.0421 \\ 0 & 0 & 0.0000 & 0.0004 & 0.2390 & 0.0038 & 0.0000 & 0.0000 & 0.2608 \\ 0 & 0 & 0 & 0.0031 & 0.0000 & 0.0000 & 0.2156 & 0.2040 & 0.0000 \\ 0 & 0 & 0 & 0 & 0.0646 & 0.0000 & 0.2365 & 0.0000 & 0.0000 \\ 0 & 0 & 0 & 0 & 0 & 0.0000 & 0.0004 & 0.0000 & 0.0045 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.0000 & 0.0000 & 0.2059 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0024 & 0.0000 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0000 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

It only makes sense to look at the values of the average affinity matrix when the stationarity of the chain has been reached. We monitored the entries of the above affinity matrix, Figure 7.12 shows the histograms and ergodic averages for entries A(1,6), A(2,9) and A(3,8). The initial values monitored for the chain show more variation and towards the last iterations the values show more stability. We observed a slightly larger variation where there are jumps in dimension, compared to the values of the sequence with the same dimension. The entries $A(i,j)$ with very small values showed more variability but in general, the matrix shows a stable behaviour. We display the matrix A graphically as a dendogram in Figure 7.14 (b). Notice that the dendogram



$A(1,6)$ $A(2,9)$ $A(3,8)$

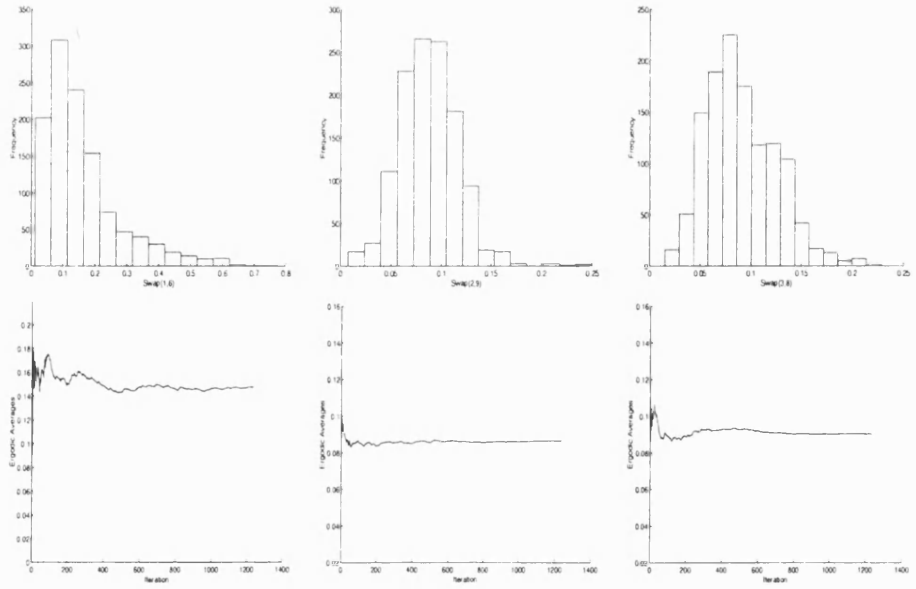
Figure 7.12: Histograms and ergodic averages for the affinity matrix: Old Faithful data, model I.

is built using $1 - A$ as a dissimilarity matrix.

The results obtained for the swap matrix are also displayed graphically as a dendogram, see Figure 7.14(a). Here again, notice that the dendogram is built using the dissimilarity matrix $1 - S$, where S is the swap matrix.

Once again it only makes sense to look at the values of the swap matrix when the stationarity of the chain has been reached. The plots for the proportion of swapped observations for the pairs $S(1,6)$, $S(2,9)$ and $S(3,8)$ are given in Figure 7.13. Here we observe more variation at the beginning of the sampler than that observed for the affinity matrix. However, variation of the swaps when dimension jumps are made shows less effects than with the affinity matrix. The entries with smaller values showed more variability but in general the matrix tends to stabilise after the first 600 monitored iterations.

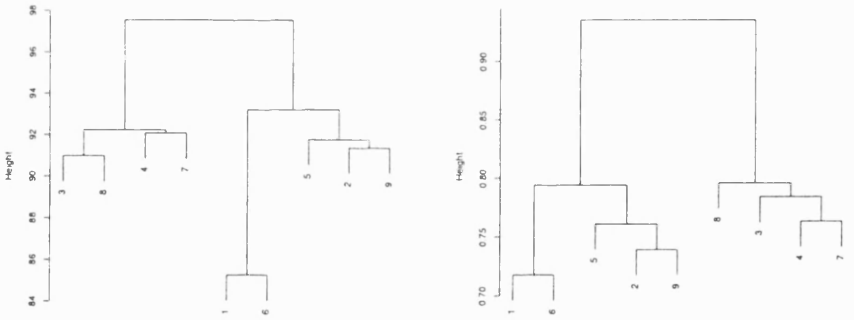
From the dendograms in Figure 7.14 we consider two groups by looking at the longest branches of the tree. The first group is described by a submixture of components (1,6,5,2,9) and the other by the submixture (3,8,4,7). In Figure 7.11 (a), they correspond to the group away from the origin (components in this group from bottom to top: yellow, black, red, cyan, blue) and to the group closer to the origin (components in this group from top to bottom: blue, red, magenta, green) respectively. From the



$S(1,6)$ $S(2,9)$ $S(3,8)$

Figure 7.13: Histograms and running means for three entries of the swap matrix in the Old Faithful data.

Figure 7.14 (b) we verify that the merged components are close together.



(a) (b)

Figure 7.14: Model I: Old Faithful data for a 9-component mixture. (a) Swap matrix. (b) Affinity matrix

The classification for the observations into these two groups based on the allocation vector, which reflects the posterior probabilities of the observations belonging to each of the groups in the mixture model, is shown in Figure 7.15. We calculated the frequency over all iterations of each observation belonging to different components and place it into the group to which it was allocated in the largest proportion.

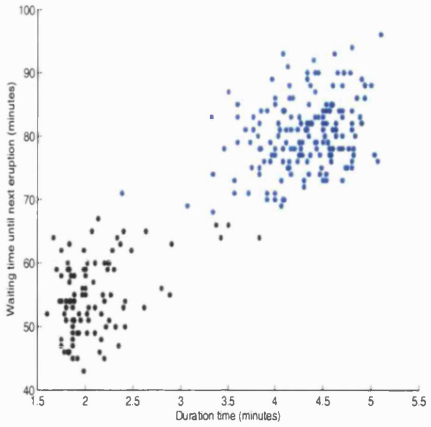


Figure 7.15: Model I: Classification for the Old Faithful data into two groups.

Results for the number of groups and the classification for the Old Faithful data based on the eight-component mixture were the same as the results described above.

Model II

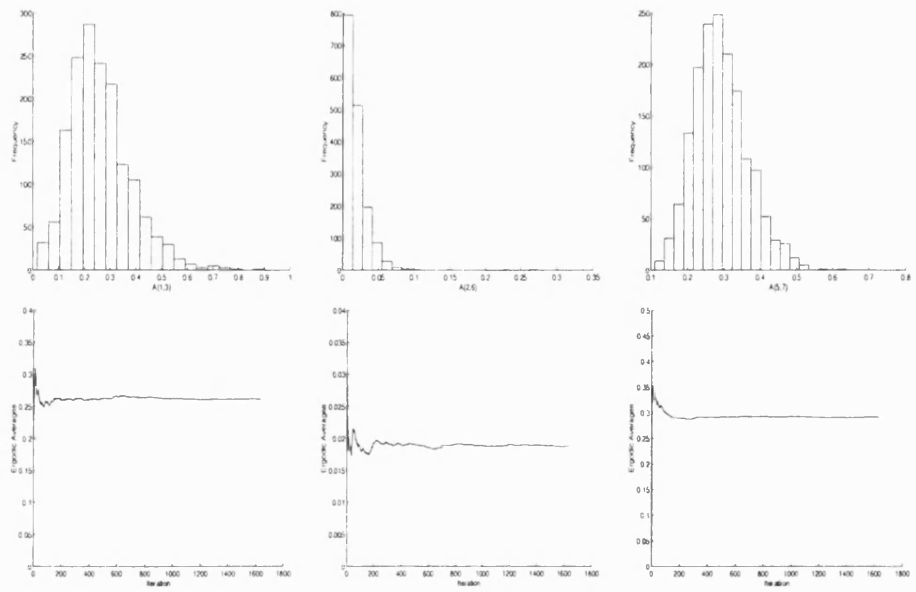
The average affinity matrix A for the seven-component mixture model fitted for model II is given below.

$$A = \begin{pmatrix} 0 & 0.0001 & 0.2640 & 0.0051 & 0.0002 & 0.1432 & 0.0214 \\ 0 & 0 & 0.0000 & 0.4056 & 0.0000 & 0.0190 & 0.0000 \\ 0 & 0 & 0 & 0.0000 & 0.0214 & 0.0019 & 0.4472 \\ 0 & 0 & 0 & 0 & 0.0000 & 0.3059 & 0.0000 \\ 0 & 0 & 0 & 0 & 0 & 0.0000 & 0.2947 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.0000 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

The affinity matrix for this model is more variable in the first iterations of the monitored period for some entries, see for example $A(2,6)$ in Figure 7.16. We show results for entries $A(1,3)$, $A(2,6)$ and $A(5,7)$. In general, after 200 iterations the behaviour stabilised and the conclusions drawn for the cluster analysis did not change when the first 200 iterations were excluded from the analysis.

A graphical display of matrix A is given as a dendogram in Figure 7.18 (b) using $1 - A$ as a dissimilarity matrix.

The results for the swap matrix for this model are given in Figure 7.18 (a). The behaviour of the swap matrix in this example showed that the proportion of observa-



$A(1,3)$ $A(2,6)$ $A(5,7)$

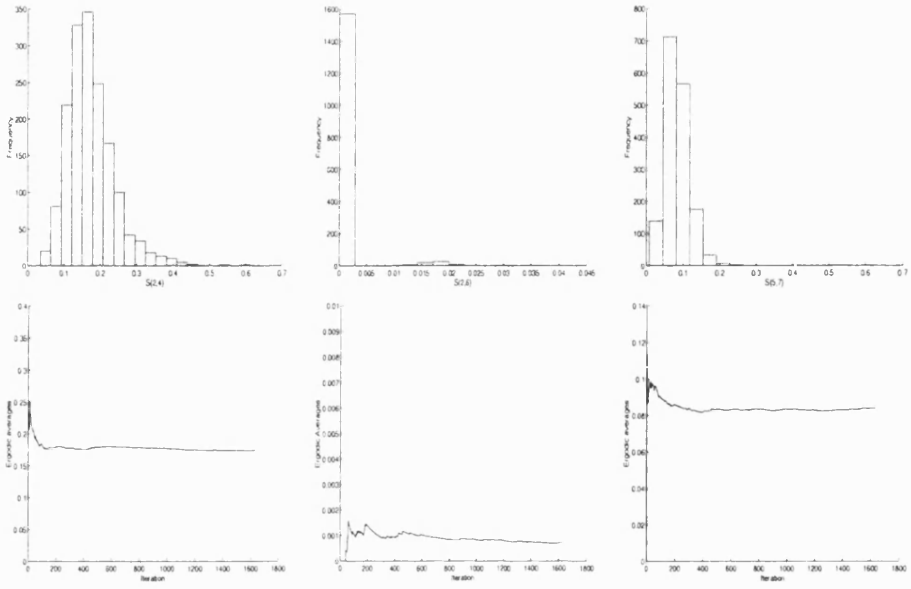
Figure 7.16: Histograms and ergodic averages for the affinity matrix: Old Faithful data, model II.

tions exchanged for some pairs of components is very small, see for example the results for the pair (2,6) in Figure 7.17, again it shows more variability at the beginning of the monitored period.

As with model I we suggest two groups to describe the data set. The first group is described by a submixture of components $(3, 7, 1, 5)$ and the second one is described by a submixture of components $(2, 4, 6)$. Here components 1 and 5 exchanged a smaller proportion of observations compared with the rest. In Figure 7.11 (c) the first groups corresponds to the group away from the origin, from bottom to top components: blue, red, yellow and magenta. The second group is closer to the origin, the components in this group from top to bottom: black, green and cyan.

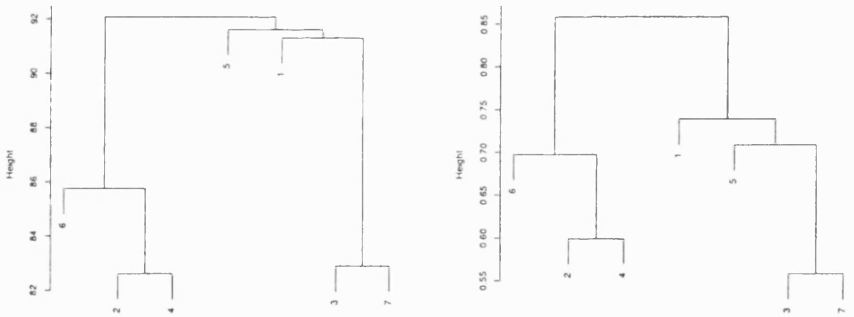
The dendogram for the affinities given in Figure 7.18 (b) shows that the components that are exchanging a large proportion of observations are close together. The corresponding classification based on the allocation vector is given in Figure 7.19.

Using a submixture of components to describe the Old Faithful data with models I and II we obtain two groups. The classification for model II is perhaps closer to what one would expect from the visual inspection of the data.



$S(2,4)$ $S(2,6)$ $S(5,7)$

Figure 7.17: Histograms and running means for three entries of the swap matrix in the Old Faithful data.



(a) (b)

Figure 7.18: Model II: Old Faithful data for a 7-component mixture. (a) Swap matrix. (b) Affinity matrix

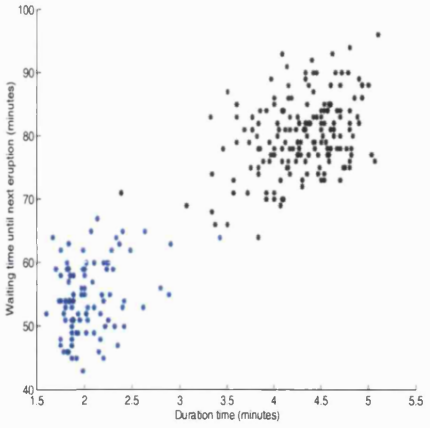


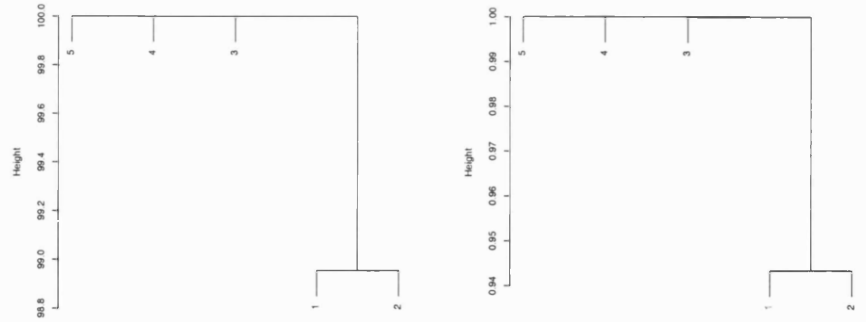
Figure 7.19: Model II: Classification for the Old Faithful data into two groups.

Example 2: Ruspini data.

The Ruspini data set is described by a five-component mixture with the highest posterior probability for both models. The affinity and the swap matrices showed very similar behaviour to that of the corresponding matrices in the Old Faithful data example. There is slightly more variability at the beginning of the iterations and it stabilises towards the end of the sampled iterations, depending heavily on the stationarity of the chain. The entries of the matrices which have very small values showed in general more variability. We omit the results for brevity.

Model I

The swap and affinity matrices for model I are displayed graphically as a dendogram in Figure 7.20 (a) and (b) respectively. In this example, we find that most components do not exchange observations, except for components 1 and 2 which exchange a small proportion of observations. From the affinity matrix we observe that the densities are not close together. In this case therefore we might suggest not to merge any components, and let each components represent a group. The classification derived from the allocation vector is given in Figure 7.21



(a) (b)

Figure 7.20: Model I: Ruspini data for a 5-component mixture. (a) Swap matrix. (b) Affinity matrix

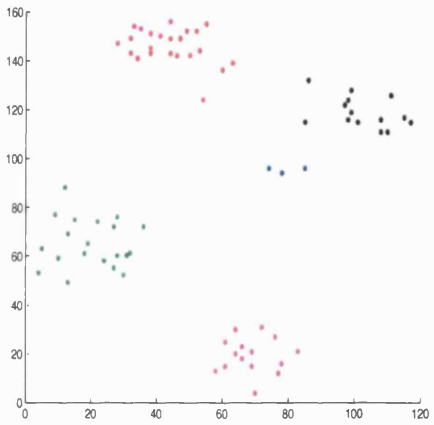


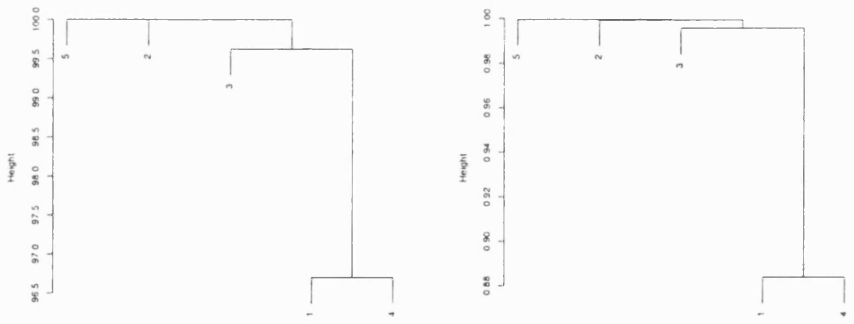
Figure 7.21: Model I: Classification for the Ruspini data into five groups.

Model II

The swap and affinity matrices for model II are displayed graphically as a dendogram in Figure 7.22 (a) and (b) respectively. With this model, components 1 and 4 exchange a larger proportion of observations, compared to components 1 and 2 in model I. However, from the dendogram in Figure 7.22 (b), we observe that the distance between this pair of components is large. Hence, we suggest not merging any components.

The classification based on the allocation vector is the same as the one obtained for model I, see Figure 7.21. If one decided to merge components 1 and 4, the component in blue would be merged with the black component.

The use of submixtures in this case gives the same results as the ones obtained with



(a) (b)

Figure 7.22: Model II: Ruspini data for a 5-component mixture. (a) Swap matrix. (b) Affinity matrix

unrestricted models using RJMCMC (with a data informed split/combine move) and BDMCMC samplers.

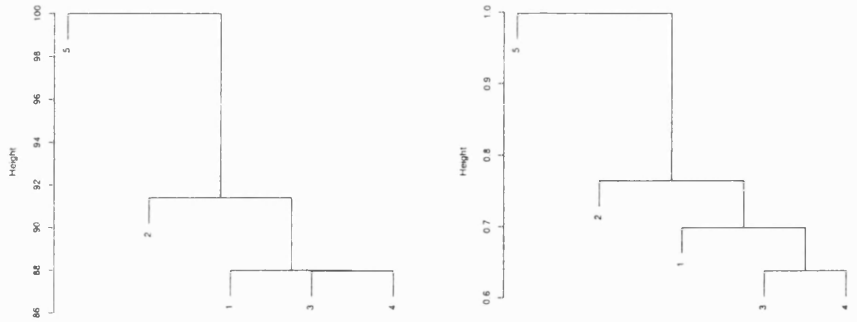
Example 3: Iris data.

The first model fits a five-component mixture with the highest posterior probability to the Iris data set. The second model fits a four-component mixture with the highest posterior probability. The swap and affinity matrices show similar behaviour to that observed in the Old Faithful example. In general, for both models, the entries in the affinity matrix that are more stable are the ones that correspond to pairs of components that are close together. For the swap matrix, the entries with the larger proportion of exchanged observation are also more stable compared to the ones that only exchange observations occasionally.

Model I

With this model the Iris data set is described with a five-component mixture with the highest posterior probability. The swap and affinity matrices for model I are displayed graphically in Figure 7.23 (a) and (b) respectively. In this example, the component labelled as five does not exchange observations with any other component. It corresponds to the first species, the *setosa* species. From Figure 7.23 (a), we find either that component two could be considered as another separated group or else we could place all the remaining components in only one group. From the affinity matrix, Figure 7.20 (b) we observe that component 2 is separated from components 1, 3 and 4.

Therefore, we could consider three groups, one formed by components 1, 3 and 4 and the other two groups corresponding to components 5 and 2.



(*a*) (*b*)

Figure 7.23: Model I: Iris data for a 4-component mixture. (a) Swap matrix. (b) Affinity matrix

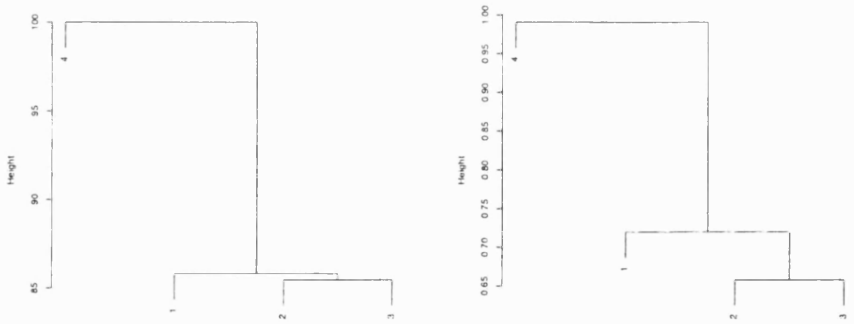
The classification based on the allocation vector places the first 50 observations in component 5, they correspond to the *setosa* species. The second group has only 9 observations (106, 108, 110, 118, 119, 123, 131, 132, 136), all belonging to the third species, the *versicolor* species. The remaining observations are placed all together in the last group.

Model II

The swap and affinity matrices for model II are displayed graphically in Figure 7.24 (a) and (b) respectively.

With model II the Iris data are described with a four-component mixture. From the swap matrix, we find that the component labelled four does not exchange observations with any other component. The remaining three components exchange a large proportion of observations. However, from the affinity matrix in Figure 7.24 (b) we see that component one could be considered as separated from the rest.

The classification based on the allocation vector shows that component number 4 corresponds to the first species, the *setosa* species. The observations that belong to the second species, the *versicolor* species, are allocated to component 3 (26 observations) and to component 2 (24 observations). The last species, the *virginica*, has 23 observations in component 1 and 27 observations in component 2. If component one is considered as a separated species, there are 23 observations allocated to this group, all



(a) (b)

Figure 7.24: Model II: Iris data for a 4-component mixture. (a) Swap matrix. (b) Affinity matrix

belonging to the *virginica* species.

In the Iris data set, two of the groups are very similar and it is difficult to separate them. Using the submixtures we obtain two groups but there is additional information on the possibility of having three groups. Further analysis on the second group could be carried out.

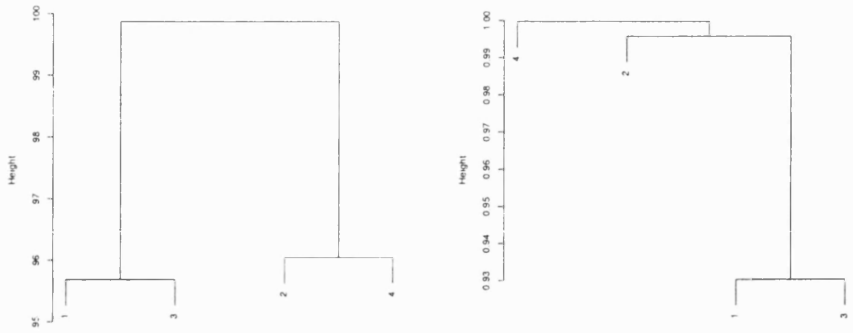
Example 4: Lubischew's beetle data.

This data set is described with a four-component mixture with the highest posterior probability for models I and II. No significant differences in the behaviour of the swap and the affinity matrices with respect to what we have pointed out in previous examples were found.

Model I

The swap and affinity matrices for model I are displayed graphically in Figures 7.25 (a) and (b) for a four-component mixture, the model with the highest posterior probability. Since a five-component mixture also has a large posterior probability, we analysed this model to verify the results are consistent in terms of the groups they define. The swap and affinity matrix for the five-component mixture are given in Figure 7.26 (a) and (b) respectively.

In the four-component model, components 1 and 3 exchange a few observations and also components 2 and 4, see Figure 7.25 (a). However, from the affinity matrix, see Figure 7.25 (b), components 2 and 4 are separated compared to components 1 and 3



(a) (b)

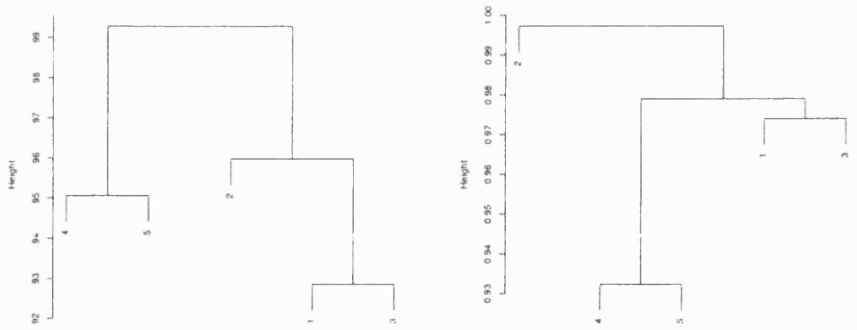
Figure 7.25: Model I: Lubischew's beetle data for a 4-component mixture. (a) Swap matrix. (b) Affinity matrix

which are close to each other. Notice that the distances between pairs of components for the Lubischew's beetle data are larger than the ones we have observed in other examples. To define the groups in this data set, we must look at the distances relative to this particular example. We suggest to consider the components as separate groups or at most merge components 1 and 3 into one group.

The classification obtained from the allocation vector places 18 observations in component 1, all belonging to the second species, the *heikertingeri*. Component 2 has 20 observations all from the first species, the *concinna*, component 3 has 13 observations that belong to the second species, the *heikertingeri* and component 4 has 23 observations in it, 22 from the third species the *heptapotamica* and one that belongs to the first species the *cocinna* (observation 6).

In the five-component mixture, from the swap matrix (Figure 7.26 (a)), the model would place components 1 and 3 into one group and components 4 and 5 into another group. From affinity matrix we find that component 2 is not close to the other components and therefore we could separate it as the third group. Here again, the distances between components are larger than those we have observed in other examples.

The classification based on the allocation vector places in the first group 22 observations mainly from the first species *conccina*, although it has observation 25 from the *heikertingeri* species and observation 53 which belongs to the third species *heptapotamica*. The second group has the remaining 30 observations from the second species, the *heikertingeri* species, allocated as follows 12 to component 4 and 18 to



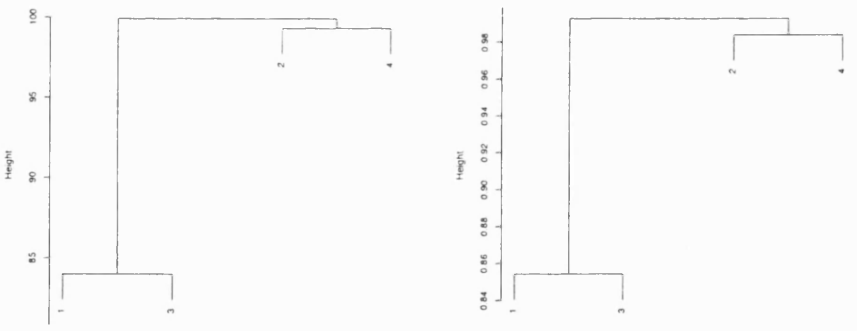
(a) (b)

Figure 7.26: Model I: Lubischew's beetle data for a 5-component mixture. (a) Swap matrix. (b) Affinity matrix

component 5. The last group has 21 observations from the third species *heptapotamica* and observation 6 from the *conccina* species.

Model II

The swap and affinity matrices for model II are displayed graphically in Figure 7.27 (a) and (b) respectively.



(a) (b)

Figure 7.27: Model II: Lubischew's beetle data for a 4-component mixture. (a) Swap matrix. (b) Affinity matrix

This model merges components 1 and 3 into one group. Components 2 and 4 exchanged a few observations but from the affinity matrix in Figure 7.27 (b) we see that they are not close together, so we leave them as a separated group.

The classification based on the allocation vector places the 21 observations from the

concinna species in component labelled as 4. The 31 observations from the *heikertingeri* species are placed 18 in component 1 and 13 in component 3. The last group, which corresponds to component 2 has the 22 observations from the *heptapotamica* species.

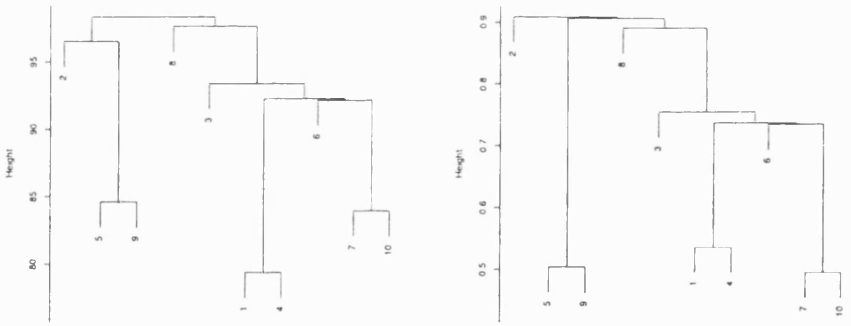
From the information of all models, we conclude that there are three groups in the Lubischew's data set. In particular, the classification obtained from model II separated the data efficiently.

Example 5: Simulated data.

The simulated data set is described with a 10-component mixture with the highest posterior probability with model I. Model II fits a six-component mixture with the highest posterior probability. The behaviour of the swap and the affinity matrices is similar to the behaviour described in previous examples.

Model I

The swap and affinity matrices for model I are displayed graphically in Figure 7.28 (a) and (b) respectively.



(a) (b)

Figure 7.28: Model I: Simulated data for a 10-component mixture. (a) Swap matrix. (b) Affinity matrix

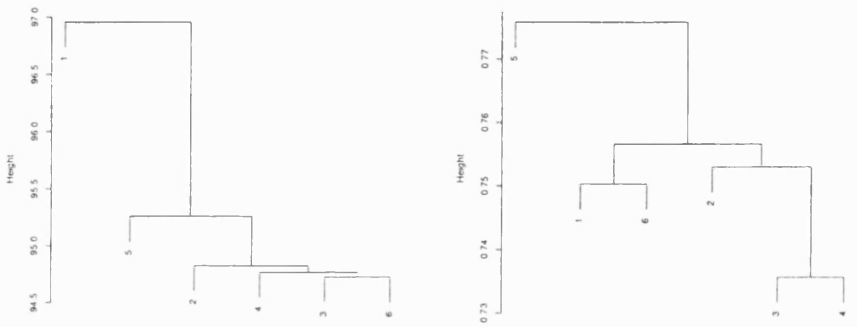
From the swap matrix in Figure 7.28 (a) we have that the pairs of components $(1, 4)$, $(5, 9)$ and $(7, 10)$ have the largest proportions of exchanged observations. Figure 7.28 (b) shows that these are also the closest pairs of components. These pairs are merged into one group. The rest of the components are considered as separated groups. We have in total 7 groups, six are centered near the means that were used to simulate the data. Looking at the sampled mean vectors the group that corresponds to component

labelled as 6, this component has is centered around point $(9.5, 9.5, 9.5)$.

The classification based on the allocation vector places in each component 38, 43, 72, 34, 96, 45, 38, 21, 24 and 89 observations respectively. The corresponding groups they define have 72, 72, 127, 21, 43, 120 and 45 observations, the latter corresponds to the component labelled as 6. The number of misclassified observations for each of the six groups are 12, 20, 17, 3, 5 and 4 respectively. The total number of misclassified observations is 61, from which 45 are placed in the seventh group.

Model II

The swap and affinity matrices for model II are displayed graphically in Figure 7.29 (a) and (b) respectively.



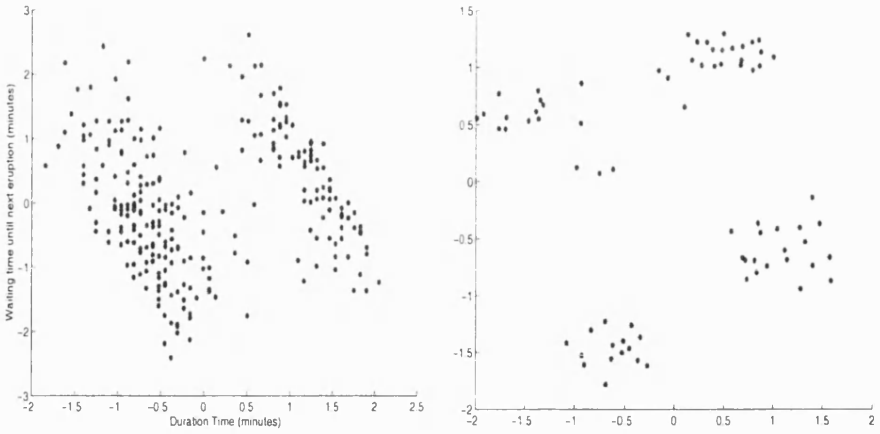
(*a*) (*b*)

Figure 7.29: Model II: Simulated data for a 6-component mixture. (a) Swap matrix. (b) Affinity matrix

In the six-component mixture, from the swap matrix in figure 7.29 (a) we consider components 1 and 5 as separated groups. Components $2, 3, 4$ and 6 exchange very few observations. From the affinity matrix in Figure 7.29 (b) we have that these components are separated and that every pair has approximately the same distance to each other. Hence, we consider each component as a group.

The classification based on the allocation vector places in each group 83, 88, 149, 20, 40 and 120 observations respectively. There are 56 misclassified observations in total mainly from the second and the third groups, which are dense groups. The number of misclassified observation per group are 9, 14, 17, 4, 8 and 4 respectively.

From the information obtained from both models we conclude that there are six groups in the simulated data set.



(a) (b)

Figure 7.30: *(a)* Rescaled Old Faithful data. *(b)* Rescaled Ruspini data.

7.5.2 Sensitivity to rescaling

Rescaling of the data will clearly modify the mixture model using the restrictions described in this chapter. In this section we illustrate the sensitivity to rescaling of the data through the Old Faithful and Ruspini data sets using model I. We transformed these data so that they are centered at zero and they have covariance matrix equal to the identity matrix, see Figure 7.30.

For the Old Faithful data, the sampler fits a seven-component mixture model with the highest posterior probability, in contrast with the nine-component mixture fitted to the original data. There was a large difference in scale and range in the observed variables in the original data set. Figures 7.31 (a)-(d) show the corresponding analysis described in this chapter. In Figure 7.31 (a) the correspondence of number and color for the sampled components is as follows: (component-colour) 1-blue, 2-red, 3-green, 4-magenta, 5-cyan, 6-black and 7-yellow. From the dendograms in Figures 7.31 (b) and (c) we would merge the components into two groups, the first including components 3, 4, and 7 and the second group includes components 1,2, 5 and 6. The resulting classification is consistent with results obtained from the nonrestricted models.

Results for the Ruspini data set show that a 4-component mixture is fitted with the highest posterior probability in contrast to the 5-component mixture fitted to the original data. The results are shown in Figures 7.32 (a)-(d) and from the affinity and swap matrices we would not merge any components. This analysis will define four groups in this data set and the corresponding classification is given in Figure 7.32 (d).

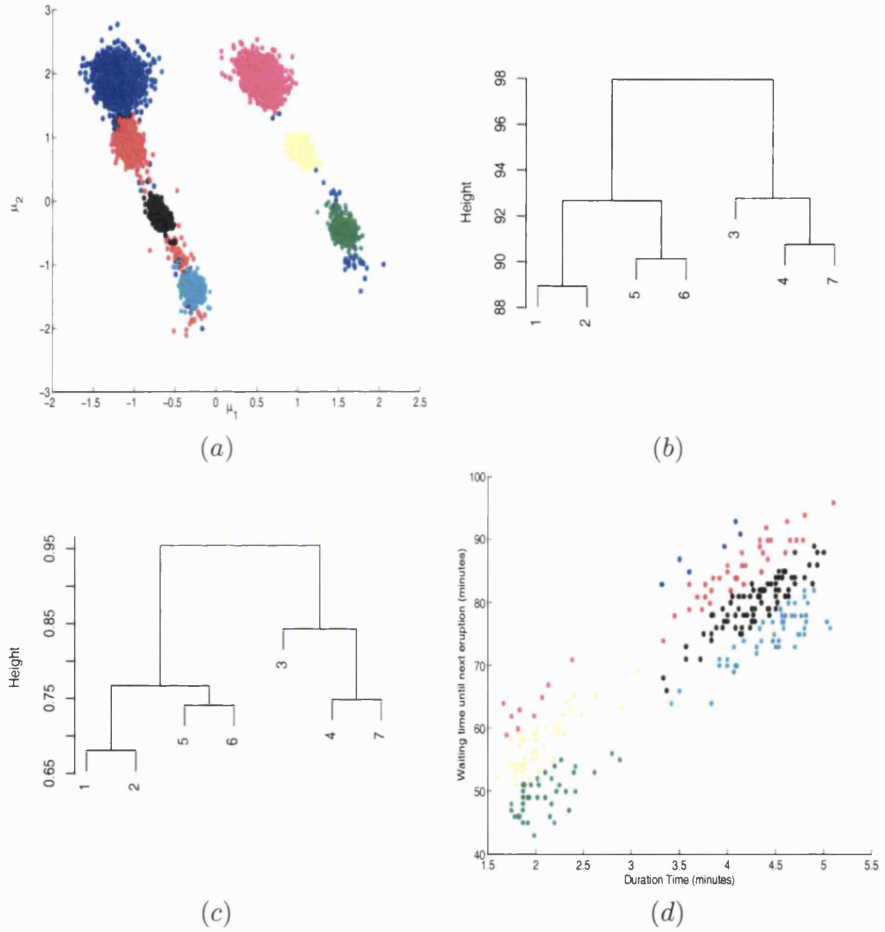


Figure 7.31: Old Faithful data. *(a)* Sampled mean values for a 7-component mixture after removing the label switching. *(b)* Swap matrix. *(c)* Affinity matrix. *(d)* Classification of the Old Faithful data.

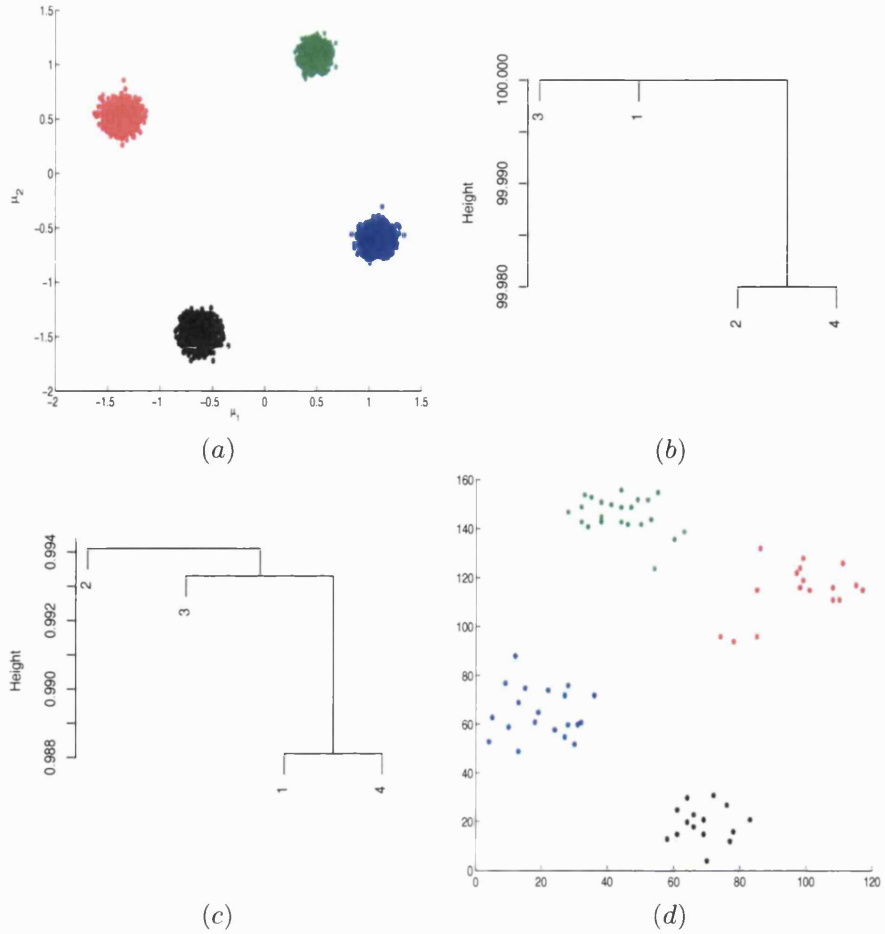


Figure 7.32: Ruspini data. *(a)* Sampled mean values for a 4-component mixture after removing the label switching. *(b)* Swap matrix. *(c)* Affinity matrix. *(d)* Classification of the Ruspini data.

In this case the rescaling of the data gives a smaller number of components but the classification is consistent with the classification obtained with the nonrestricted and restricted models fitted to the original data.

Based on the results obtained in this section we consider that rescaling of the data is advisable before fitting the restricted models proposed in this chapter. We could make use of external criteria to determine if rescaling is needed. The results showed that a smaller number of components is required to described the data when these are in agreement with the model. Improvement is also observed in terms of classification of the data.

In the cluster analysis based on submixtures, the most significant aspect to consider is the proportion of observations that were exchanged between components but it is also necessary to verify that those components are closer to each other than other pairs in the fitted model. That will determine which of the pairs that are near each other should be merged helping to identify some overlapping clusters in a more efficient way. It avoids merging pairs of components which have swapped observations mainly when the sampled values for the parameters of the components are in the tails of the corresponding distributions and indicate when a large proportion of observations is swapped between components which are very close together. However, we will always have the problem of combining groups that are very close to each other and have many observations in the *boundaries* of the two components. In this case, an important number of observations would be exchanged between these components and the criteria would suggest to consider them as only one group.

Much of the time, using the restricted covariance structure does not result in the production of many more components than groups. However, in general, it provides useful information on the number of groups and classification. Models I and II could be used to describe a data set and the conclusions on the final number of groups found in the data could be made using the information from both models. The final classification of the data could be done in a similar way, using information from both models, reducing the number of misclassified data and finding the observations that are harder to classify. Finally, we could compare the results obtained from these restricted models with those from the non-restricted model getting an idea of the robustness of the clustering methodology.

CHAPTER 8

Discussion and future work

Model-based cluster analysis using a finite mixture of normal distributions as the underlying model has been frequently used and discussed in literature. In this thesis we have looked at a Bayesian model-based clustering using mixtures of multivariate normal distributions where the number of components is considered unknown and regarded as an additional parameter of the model, therefore subject to estimation. We are also looking at problems where there are several features measured for each observation, that is, our variables correspond to p-dimensional vectors.

The estimation of the parameters in the model was possible using trans-dimensional MCMC samplers such as the RJMCMC and the BDMCMC. It is difficult to evaluate the performance of a clustering methodology. One way of assessing this is to examine the results of the cluster analysis with examples where the groups are previously known ensuring that no groups are found in data sets where there are no groups present. We have followed this line describing the results for a set of examples that included simulated data as well as known data sets. The general conclusions will be discussed in this chapter.

In Chapter 3 we considered each component of the mixture model as a group and used a RJMCMC sampler to carry out estimation. We found that the sampler needs long runs in order to stabilise and is effective in detecting the structure in the data when the groups are well separated and also when the model conforms with the data. The acceptance rates of the split/combine moves are small and therefore we explored some data driven split/combine moves in Chapter 4 based on principal components and minimum spanning trees. The acceptance rates are improved in some examples and the RJMCMC sampler could be modified to include a variety of split/combine moves

to give the sampler the opportunity to explore different areas of the parameter space.

In Chapter 5 we used a BDMCMC sampler to carry out inference. This sampler is as computationally demanding and complex as the RJMCMC but it was implemented in a more straightforward way. The sampler also needs long runs to stabilise but it explores more of the parameter space and therefore performs better than the RJMCMC in some examples such as the Iris data set.

The posterior distribution of the number of components of the mixture model is highly sensitive to the model assumptions. However the conclusions on the clustering and classification in general remain unchanged. The mixing of the sampler over the number of components is often poor, and assessing the convergence is not straightforward. In Chapter 6 we have described some recent methods to evaluate the convergence of a trans-dimensional MCMC sampler. In general we confirmed that long runs are required for the trans-dimensional samplers to converge, possibly discarding the first iterations of the outputs which were used for inference. The conclusions in terms of the clustering and classification did not change when the first few iterations of the outputs were discarded.

The efficiency and in some sense the attractive simplicity of the Gaussian mixture model is not always the best option to capture the structure of the data. We observed that the sampler often includes small weighted components or prefer a small number of highly dispersed components to accommodate departures from normality. In Chapter 7 we proposed to allow a submixture of components to represent a cluster and at the same time restricted the shapes of the multivariate normal distributions. We encouraged the use of more components than groups through models I and II and examined two criteria to merge the resulting components into submixtures which describe a single group. The first criterion takes into account the proportion of disputed observations between two components and the second criterion considers the closeness of the model components. The results were satisfactory in most cases. However, it is difficult to identify groups that overlap and which have many observations in the boundaries.

We found that a combination of the methods could be effective and very informative on the structure of the groups. The conclusions on the number of clusters and the classification of the data could be given combining the results of all fitted mixtures and submixtures of multivariate normal distributions. For example, the Old Faithful data set was described by a three-component mixture with the highest posterior

probability using the unrestricted mixture model, where one of the components was a small weighted component which had five observations allocated into it. Using the submixtures with models I and II we found that there are two groups in this data set. This indicates that in this case the small weighted component is likely to be reflecting non normality. The classification varies in each case for a very small number of observations.

The Ruspini data set had a five-component mixture with the highest posterior probability in most cases. Here we have that considering the submixtures of components as described in Chapter 7 each one of the five components of the fitted mixture represents a group and the classification of the observations remains unchanged in all cases. In this case the small weighted component could be considered as a different group. Notice that clustering of this data set gives very different results with different clustering algorithms. Peña and Prieto [47] for example found seven groups in this data set where two of those groups had only one observation allocated into them.

Lubischew's beetle data was mostly described by a three-component mixture and the observations were successfully allocated to the species to which they belong. The most difficult data set to analyse was the Iris data set. One of the species is separated in all cases and the remaining observations are either placed into one group or separated into two groups where many observations are allocated to the wrong species.

Future work includes exploring the criteria used in Chapter 7 to merge components into a submixture that defines a group. The prior assumptions were determined using an initial analysis of the data and other possibilities will be analysed. The effects of the restricted modelling of covariance matrices of the multivariate normal distributions in the mixture model will also be revised. Other ways of determining the prior distributions on the covariance matrices of the restricted mixture models will be explored, evaluating the effect of how we determined constant c. Finally, there are many different techniques to partition dendograms, some other techniques will be explored and the resulting groups will be compared to the ones obtained from looking for the longest branches.

Another interesting aspect we need to address is the procedure to identify the mixture components when the number of components is large. We have used the early iterations of the MCMC sampler to determine a modal region and the following iterations are relabelled by looking at all the possible permutations of the components.

As the number of components increases, a search of a small subset of all possible permutations could be found to be efficient.

Recently mixtures of multivariate normal distributions have been used to analyse microarray gene data, see for example in MacLachlan *et al* [44]. Bayesian model-based clustering as we have presented in this thesis could be useful in some cases to analyse microarray gene data sets. The probabilistic interpretation of the model and the potential for classification of future observations would be particulary attractive in this area, where researchers are often looking for groups of genes involved in the same biological function or associated with a particular event.

Appendix A

Jacobian for the transformation in the moment matching type split/combine move

Consider the deterministic functions defining the transformation used for the moment matching type split/combine moments given by equations (3.23). Using the equations (3.24), when $p = 1$, the jacobian is given by the matrix

$$\begin{array}{cccccc}
u & 0 & 0 & 1-u & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 \\
0 & \frac{-v}{2}\sqrt{\frac{(1-u)}{u\sigma_*^2}} & \frac{t}{u} & 0 & \frac{v}{2}\sqrt{\frac{u}{(1-u)\sigma_*^2}} & \frac{(1-t)}{(1-u)} \\
w_* & \frac{v}{2}\sqrt{\frac{\sigma_*^2}{u^3(1-u)}} & \frac{-\sigma_*^2 t}{u^2} & -w_* & \frac{v}{2}\sqrt{\frac{c}{u(1-u)^3}} & \frac{\sigma_*^2*(1-t)}{(1-u)^2} \\
0 & -\sqrt{\frac{(1-u)\sigma_*^2}{u}} & 0 & 0 & \sqrt{\frac{v\sigma_*^2}{(1-u)}} & 0 \\
0 & 0 & \frac{\sigma_*^2}{u} & 0 & 0 & \frac{-\sigma_*^2}{(1-u)}
\end{array}$$

The determinant for the jacobian was computed in Maple and the simplified expression is

$$\frac{w_*\sigma_*^3}{[u(1-u)]^{3/2}}.$$

For the case where $p = 2$, the jacobian of the transformation corresponds to the

following 12×12 matrix:

$$
\begin{array}{cccccc}
u & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & \frac{-v_1}{2}\sqrt{\frac{(1-u)}{u\sigma_{11*}^2}} & 0 & \frac{t_{11}}{u} & 0 & 0 \\
0 & 0 & \frac{-v_2}{2}\sqrt{\frac{(1-u)}{u\sigma_{22*}^2}} & 0 & \frac{t_{22}}{u} & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{t_{12}\sqrt{t_{11}t_{22}}}{u^2} \\
w_* & \frac{v_1}{2}\sqrt{\frac{\sigma_{11*}^2}{(1-u)u^3}} & \frac{v_2}{2}\sqrt{\frac{\sigma_{22*}^2}{(1-u)u^3}} & \frac{\sigma_{11*}^2 t_{11}}{u^2} & \frac{\sigma_{22*}^2 t_{22}}{u^2} & \frac{\sigma_{12*} t_{12}\sqrt{t_{11}t_{22}}}{2u^3} \\
0 & -\sqrt{\frac{(1-u)\sigma_{11*}^2}{u}} & 0 & 0 & 0 & 0 \\
0 & 0 & -\sqrt{\frac{(1-u)\sigma_{22*}^2}{u}} & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{\sigma_{11*}^2}{u} & 0 & \frac{\sigma_{12*} t_{12}\sqrt{t_{22}}}{2\sqrt{t_{11}}} \\
0 & 0 & 0 & 0 & \frac{\sigma_{22*}^2}{u} & \frac{\sigma_{12*} t_{12}\sqrt{t_{11}}}{2\sqrt{t_{22}}} \\
0 & 0 & 0 & 0 & 0 & \frac{\sigma_{12*}\sqrt{t_{11}t_{22}}}{u^2} \\
 & & & \cdots & & \\
(1-u) & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & \frac{v_1}{2}\sqrt{\frac{u}{(1-u)\sigma_{11*}^2}} & 0 & \frac{(1-t_{11})}{(1-u)} & 0 & 0 \\
0 & 0 & \frac{v_2}{2}\sqrt{\frac{u}{(1-u)\sigma_{22*}^2}} & 0 & \frac{(1-t_{22})}{(1-u)} & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{(1-t_{12})\sqrt{(1-t_{11})(1-t_{22})}}{(1-u)^2} \\
-w_* & \frac{v_1}{2}\sqrt{\frac{\sigma_{11*}}{(1-u)^3 u}} & \frac{v_2}{2}\sqrt{\frac{\sigma_{22*}}{(1-u)^3 u}} & \frac{\sigma_{11*}(1-t_{11})}{(1-u)^2} & \frac{\sigma_{22*}(1-t_{22})}{(1-u)^2} & \frac{\sigma_{12*}(1-t_{12})\sqrt{(1-t_{11})(1-t_{22})}}{2(1-u)^3} \\
0 & \sqrt{\frac{u\sigma_{11*}}{(1-u)}} & 0 & 0 & 0 & 0 \\
0 & 0 & \sqrt{\frac{(u\sigma_{22*}}{(1-u)}} & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{-\sigma_{11*}}{(1-u)} & 0 & \frac{-\sigma_{12*}(1-t_{12})\sqrt{1-t_{22}}}{-2\sqrt{1-t_{11}}(1-u)} \\
0 & 0 & 0 & 0 & \frac{-\sigma_{22*}}{(1-u)} & \frac{-\sigma_{12*}(1-t_{12})\sqrt{1-t_{11}}}{-2\sqrt{1-t_{22}}(1-u)} \\
0 & 0 & 0 & 0 & 0 & \frac{-\sigma_{12*}\sqrt{(1-t_{11})(1-t_{22})}}{(1-u)^2}
\end{array}
$$

The determinant for the jacobian in this case is given by

$$\frac{w_* \sigma_{11*}^3 \sigma_{22*}^3 \sigma_{12*} (t_{11} t_{22} (1-t_{11})(1-t_{22}))^{1/2}}{[u(1-u)]^{8/2}}.$$

When the dimension is increased to $p = 3$, the dimension of the matrix is increased to 20×20, the determinant for the jacobian in this case is given by

$$\frac{w_* \sigma_{11_*}^3 \sigma_{22_*}^3 \sigma_{33_*}^3 \sigma_{12_*} \sigma_{13_*} \sigma_{23_*} (t_{11} t_{22} t_{33} (1 - t_{11})(1 - t_{22})(1 - t_{33}))^{2/2}}{[u(1-u)]^{15/2}}.$$

However, the case when $p = 4$, which produces a jacobian of dimensions 30×30 was not computed by Maple. From the general form of the matrices we conjecture that the determinants will continue to have the same form and the exponents change as dimension increases, so they are expressed in terms of p to get the general form

$$|\mathbf{J}| = \frac{w_{j_*} \prod_{i=1}^{p} (\sigma_{ii_{j_*}}^3 (t_{ii}(1 - t_{ii}))^{(p-1)/2}) \prod_{i=1}^{p} \prod_{j=i+1}^{p} \sigma_{ij_{j_*}}}{(u(1-u))^{(2p^2+p)/2}}.$$

The last expression was verified by direct evaluation for some combine moves in the Iris example. The Jacobian for the move selected had very small values. The maximum value calculated was $1.2542e - 008$ and there was a computation error of $4e - 012$. Therefore, there is no indication of a numerical instability which could affect the computation of the acceptance probabilities for the split-combine move.

The sampler was also run without any data for a five dimensional example, for a period of 2000000 iteration and considering a maximum number of components $k_{max} = 5$. Simulating from the prior was done first considering only the birth/death move and the results were compared to the sampler were both the birth/death and the split/combine move were included. Examining the stationary distribution before and after including the split/combine move, no changes were observed. Therefore the new move type does not introduce any problems through a problem in the jacobian.

Appendix B

Affinity between two multivariate normal densities

The Bhattacharyya distance [6] also known as the affinity between two distributions P_1 and P_2, with corresponding densities p_1, p_2 is given as

$$A(p_1, p_2) = \int_{\mathbf{y}} \sqrt{p_1}\sqrt{p_2}\, d\mathbf{y}. \tag{B.1}$$

We calculate the affinity between two multivariate normal distributions, $N(\mathbf{y}|\boldsymbol{\mu}_1, \Sigma_1)$ and $N(\mathbf{y}|\boldsymbol{\mu}_2, \Sigma_2)$.

$$\begin{aligned} A(p_1,p_2) &= \int_{\mathbf{y}} \left[(2\pi)^{-\frac{p}{2}}|\Sigma_1|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu}_1)^T\Sigma_1^{-1}(\mathbf{y}-\boldsymbol{\mu}_1)\right\}\right]^{\frac{1}{2}} \times \\ &\times \left[(2\pi)^{-\frac{p}{2}}|\Sigma_2|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu}_2)^T\Sigma_1^{-1}(\mathbf{y}-\boldsymbol{\mu}_2)\right\}\right]^{\frac{1}{2}} d\mathbf{y} \\ &= \int_{\mathbf{y}} (2\pi)^{-\frac{p}{2}}|\Sigma_1|^{-\frac{1}{4}}|\Sigma_2|^{-\frac{1}{4}} \exp\left\{-\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu}_1)^T(2\Sigma_1)^{-1}(\mathbf{y}-\boldsymbol{\mu}_1)\right\} \times \\ &\times \exp\left\{-\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu}_2)^T(2\Sigma_2)^{-1}(\mathbf{y}-\boldsymbol{\mu}_2)\right\} d\mathbf{y}. \end{aligned} \tag{B.2}$$

Recall the following property of gaussian densities: the product of two gaussian densities is proportional to another gaussian density

$$N(\mathbf{y}|\boldsymbol{\mu}_1, \Sigma_1) \times N(\mathbf{y}|\boldsymbol{\mu}_2, \Sigma_2) \propto N(\mathbf{y}|\boldsymbol{\mu}_*, \Sigma_*), \tag{B.3}$$

where $\mu_* = \Sigma_*(\Sigma_1^{-1}\mu_1 + \Sigma_2^{-1}\mu_2)$ and $\Sigma_* = (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}$.

Rewriting A in expression B.2 to be able to use the above property we get

$$\begin{aligned} A(p_1,p_2) &= (2\pi)^{\frac{p}{2}}|\Sigma_1|^{-\frac{1}{4}}|\Sigma_2|^{-\frac{1}{4}}|2\Sigma_1|^{\frac{1}{2}}|2\Sigma_2|^{\frac{1}{2}} \times \\ &\times \int_{\mathbf{y}} (2\pi)^{-\frac{p}{2}}|2\Sigma_1|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(\mathbf{y}-\mu_1)^T(2\Sigma_1)^{-1}(\mathbf{y}-\mu_1)\right\} \times \\ &\times (2\pi)^{-\frac{p}{2}}|2\Sigma_2|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(\mathbf{y}-\mu_2)^T(2\Sigma_2)^{-1}(\mathbf{y}-\mu_2)\right\} d\mathbf{y}. \quad \text{(B.4)} \end{aligned}$$

Rewriting expression B.4 in terms of a normal distribution with mean vector and covariance matrix

$$\begin{aligned} \mu_* &= \Sigma_*((2\Sigma_1)^{-1}\mu_1 + (2\Sigma_2)^{-1}\mu_2), \\ \Sigma_* &= ((2\Sigma_1)^{-1} + (2\Sigma_2)^{-1})^{-1}, \end{aligned}$$

and adding the required normalizing constant, we get

$$\begin{aligned} A(p_1,p_2) &= (2\pi)^{\frac{p}{2}}|\Sigma_1|^{-\frac{1}{4}}|\Sigma_2|^{-\frac{1}{4}}|2\Sigma_1|^{\frac{1}{2}}|2\Sigma_2|^{\frac{1}{2}} \times \\ &\quad \times(2\pi)^{-\frac{p}{2}}|(2\Sigma_1)+(2\Sigma_2)|^{\frac{1}{2}}|\Sigma_1|^{-\frac{1}{2}}|\Sigma_2|^{-\frac{1}{2}} \times \\ &\quad \times \exp\left\{-\frac{1}{2}\left(\mu_1^T\Sigma_1^{-1}\mu_1 + \mu_2^T\Sigma_2^{-1}\mu_2 - \mu_*^T\Sigma_*^{-1}\mu_*\right)\right\} \\ &= |\Sigma_1|^{-\frac{1}{4}}|\Sigma_2|^{-\frac{1}{4}}|\Sigma_*|^{\frac{1}{2}} \times \\ &\quad \times \exp\left\{-\frac{1}{4}\left(\mu_1^T\Sigma_2^{-1}\mu_1 - \mu_2^T\Sigma_2^{-1}\mu_2\right) + \frac{1}{2}\left(\mu_*^T\Sigma_*\mu_*\right)\right\}, \end{aligned}$$

where now $\mu_* = ((2\Sigma_1)^{-1}\mu_1 + (2\Sigma_2)^{-1}\mu_2)$.

In the particular case where Σ_1 and Σ_2 are diagonal matrices, the affinity between p_1 and p_2 is given by

$$A(p_1,p_2) = \prod_{i=1}^{p}\left(\frac{2\sigma_{1,i}\sigma_{2,i}}{\sigma_{1,i}^2+\sigma_{2,i}^2}\right)^{\frac{1}{2}} \exp\left\{-\frac{1}{4}\sum_{i=1}^{p}\frac{(\mu_{1,i}-\mu_{2,i})^2}{\sigma_{1,i}^2+\sigma_{2,i}^2}\right\}.$$

BIBLIOGRAPHY

[1] ANDERSON, E. The irises of the Gaspe peninsula. *Bulletin of the American Iris Society 59* (1935), 2–5.

[2] ANTONIAK, C. E. Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *Ann. Stats. 2* (1974), 1152–1174.

[3] BANFIELD, D. B., AND RAFTERY, A. E. Model-based Gaussian and non-Gaussian clustering. *Biometrics 49* (1993), 803–821.

[4] BENSMAIL, H., CELEUX, G., RAFTERY, A. E., AND ROBERT, C. P. Inference in model-based cluster analysis. *Stat. Comput. 7* (1997), 1–10.

[5] BERNARDO, J. M., AND SMITH, A. F. M. *Bayesian Theory.* New York: Wiley, 1994.

[6] BHATTACHARYYA, A. On a measure of divergence between two statistical populations defined by their probability distributions. *Bull. Calcutta Math. Soc. 35* (1943), 99–110.

[7] BOLTON, R. J., AND KRZANOWSKI, W. J. Projection pursuit clustering. *JCGS 12* (2003), 121–142.

[8] BROOKS, S. J., AND GELMAN, A. General methods fot monitoring convergence of iterative simulations. *JCGS 7* (1998), 434–455.

[9] BROOKS, S. J., AND GIUDICI, P. MCMC convergence assesment via two way ANOVA. *JCGS 9* (2000), 266–285.

[10] BROOKS, S. P., GIUDICI, P., AND PHILLIPE, A. Nonparametric convergence assessment for MCMC model selection. *JCGS 12* (2003), 1–22.

[11] BROOKS, S. P., GIUDICI, P., AND ROBERTS, G. O. Efficient construction of reversible jump MCMC. *J. R. Statist. Soc. B 65* (2003), 3–55.

[12] CAPPÉ, O., ROBERT, C. P., AND RYDÉN, T. Reversible jump, birth-and-death and more general continuous time Markov chain Monte Carlo samplers. *J. R. Statist. Soc. B 65* (2003), 679–700.

[13] CASTELLOE, J., AND ZIMMERMAN, D. Convergence assessment for reversible jump MCMC samplers. *Technical Report* (2004).

[14] CELEUX, G. *COMPSTAT 98.* Physica-Verlag, 1998, ch. Bayesian inference for mixtures: The label switching problem, pp. 227–232.

[15] CELEUX, G., AND GOVAERT, G. Gaussian parsimonious clustering models. *Pattern recognition 28* (1995), 781–793.

[16] CELEUX, G., HURN, M., AND ROBERT, C. P. Computational and inferential difficulties with mixture posterior distributions. *J. Am. Statist. Ass. 95* (2000), 957–970.

[17] COCHRAN, W., AND COX, G. *Experimental designs.* John Wiley & Sons, 1992.

[18] DASGUPTA, S., AND RAFTERY, A. E. Detecting features in spatial point processes with clutter via model-based clustering. *J. Am Statist. Ass. 93* (1998), 294–302.

[19] DELLAPORTAS, P., AND PAPAGEORGIOU, I. Multivariate mixtures of normals with an unknown number of components. *Technical Report 313, University of Iowa* (2002).

[20] DEMPSTER, A. P., LAIRD, N. M., AND RUBIN, D. B. Maximim likelihood from incomplete data via EM algorithm. *J. R. Statist. Soc. B 39* (1977), 1–38.

[21] EDWARDS, A., AND CAVALLI-SFORZA, L. L. A method for cluster analysis. *Biometrics 21* (1965), 362–375.

[22] ESCOBAR, M. D. Estimating normal means with a Dirichlet process prior. *J. Am. Statist. Ass. 89* (1994), 268–277.

[23] ESCOBAR, M. D., AND WEST, M. Bayesian density estimation and inference using mixtures. *J. Am. Statis. Ass. 90* (1995), 577–588.

[24] FERGUSON, T. S. A Bayesian analysis of some nonparametric problems. *Annals of Statistics 1* (1973), 209–230.

[25] FRALEY, C., AND RAFTERY, A. E. How many clusters? Which clustering method? Answers via model-based cluster analysis. *The Computer Journal 41* (1998), 21–29.

[26] FRALEY, C., AND RAFTERY, A. E. Model-based clustering, discriminant analysis, and density estimation. *J. Am. Statist. Ass. 97* (2002), 611–631.

[27] FRÜHWIRTH-SCHNATTER, S. Markov chain Monte Carlo estimation of classical and dynamic switching and mixture models. *J. Am. Statist. Ass. 96* (2001), 194–209.

[28] GAMERAN, D. *Markov Chain Monte Carlo.* Chapman & Hall, 1997.

[29] GELMAN, A., CARLIN, J., STERN, H., AND RUBIN, D. *Bayesian data analysis.* Chapman & Hall, 2004.

[30] GELMAN, A., AND RUBIN, D. Inference for iterative simulation using multiple sequences. *Statistical Sciences 7* (1992), 457–511.

[31] GORDON, A. D. *Classification*, 2nd. ed. Chapman & Hall, 2000.

[32] GREEN, P. J. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika 82* (1995), 711–732.

[33] GREEN, P. J. *Monographs on Statistics and Applied Probability 87. Complex Stochastic Systems.* Chapman & Hall, 2001, ch. A Primer on Markov Chain Monte Carlo, pp. 1–62.

[34] GREEN, P. J. *Highly Structured Stochastic Systems.* Oxford University Press, 2003., ch. Trans-dimensional Markov Chain Monte Carlo, pp. 1–28.

[35] GREEN, P. J., AND RICHARDSON, S. Modelling heterogeneity with and without the Dirichlet process. *Scandinavian journal of Statistics 28* (2001), 731–792.

[36] HÄRDLE, W. *Smoothing techniques with implementation in S.* Springer, New York, 1991.

[37] HARTIGAN, J. A. *Clustering algorithms.* John Wiley & Sons, 1975.

[38] HASTINGS, W. K. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika 57* (1970), 97–109.

[39] KASS, R. E., AND RAFTERY, A. E. Bayes factors. *J. Am. Statist. Ass. 90* (1995), 773–795.

[40] LIU, S. *Monte Carlo Strategies in Scientific Computing.* Springer, New York, 2001.

[41] LO, A. Y. On a class of Bayesian nonparametric estimates: I. density estimates. *Ann. Stats. 12* (1984), 351–357.

[42] LUBISCHEW, A. On the use of discriminant functions in taxonomy. *Biometrics 18* (1962), 455–477.

[43] MCLACHLAN, G., AND PEEL, D. *Finite Mixture Models.* John Wiley & Sons, 2000.

[44] MCLACHLAN, G. J., BEAN, R. W., AND PEEL, D. A mixture model-based approach to the clustering of microarray expression data. *Bioinformatics 18* (2002), 413–422.

[45] METROPOLIS, N., ROSENBLUTH, A., ROSENBLUTH, M. N., TELLER, A. H., AND TELLER, E. Equations of state calculations by fast computing machines. *Journal of Chemical Physics 21* (1953), 1087–1091.

[46] NUMELIN, E. *General Irreducible Markov chains and non-negative operators.* Cambridge Univ. Press, 1984.

[47] PEÑA, D., AND PRIETO, J. F. Cluster identification using projections. *J. Am. Statist. Ass. 96* (2001), 1433–11445.

[48] POSSE, C. Hierarchical model-based clustering for large data sets. *JCGS 10* (2001), 464–486.

[49] QUINTANA, F. A., AND IGLESIAS, P. L. Bayesian clustering and product partition models. *J. R. Statist. Soc. B 65* (2003), 557–574.

[50] RAFTERY, A. *Practical Markov chain Monte Carlo.* Chapman & Hall, 1996, ch. Hypothesis testing and model selection via posterior simulation, pp. 163–168.

[51] RICHARDSON, S., AND GREEN, P. J. On Bayesian analysis of mixtures with an unknown number of components. *J. R. Statist. Soc. B 59* (1997), 731–792.

[52] RIPLEY, B. D. Modelling spatial patterns. *J. R. Statist. Soc. B 39* (1977), 172–212.

[53] ROBERT, C., AND CASELLA, G. *Monte Carlo Statistical Methods.* Springer, New York, 1999.

[54] RUSPINI, E. H. A new approach to clustering. *Information and Control 15* (1969), 22–32.

[55] RUSPINI, E. H. Numerical methods for fuzzy clustering. *Information Sciences 2* (1970), 319–350.

[56] SAHU, S. K., AND CHENG, C. H. A fast distance based approach for determining the number of components in a mixture. *The Canadian Journal of Statistics 31* (2003), 3–22.

[57] SCHWARZ, G. Estimating the dimension of a model. *Ann. Stats. 6* (1978), 461–464.

[58] SEBER, G. A. F. *Multivariate Observations.* Jon Wiley & Sons, 1984.

[59] STEPHENS, M. Bayesian analysis of mixture models with an unknown number of components- an alternative to reversible jump methods. *Ann. Stats. 28* (2000), 40–74.

[60] STEPHENS, M. Dealing with label switching in mixture models. *J. R. Statist. Soc. B 62* (2000), 795–809.

[61] SYMONS, M. S. Clustering criteria and multivariate mixture models. *Biometrics 37* (1982), 35–43.

[62] TIBSHIRANI, R., WALTHER, G., AND HASTIE, T. Estimating the number of clusters in a data set via the gap statistic. *J. R. Statist. Soc. B 63* (2001), 441–423.

[63] TIERNEY, L. Markov chains for exploring posterior distributions. *Ann. Stats. 22* (1994), 1701–1728.

[64] VENABLES, W. N., AND RIPLEY, B. D. *Modern Applied Statistics with S-plus*, 2nd. ed. Springer, New York, 1997.

[65] WAKEFIELD, J. C., ZHOU, C., AND SELF, S. *Bayesian Statistics 7*. Oxford University Press, 2003, ch. Modelling gene expression data over time: curve clustering with informative prior distributions.

[66] WEST, M., MÜLLER, P., AND ESCOBAR, M. D. *Aspects of Uncertainty: a Tribute to D. V. Lindley*. New York: Wiley, 1994, ch. Hierarchical priors and mixture models, with applications in regression and density estimation, pp. 363–386.

[67] WU, C. F. J. On the convergence properties of the EM algorithm. *Ann. Stats. 11* (1983), 95–103.

[68] YEUNG, K. Y., FRALEY, C., MURUA, A., RAFTERY, A. E., AND RUZZO, W. L. Model-based clustering and data transformations for gene expression data. *Bioinformatics 17* (2001), 977–987.

[69] ZHIHUA, Z., KAPLUK, C., YIMING, W., AND CHIBIAO, C. Learning a multivariate Gaussian mixture model with reversible jump MCMC algorithm. *Statistics and Computing 14* (2004), 343–355.